



Universidad
Carlos III de Madrid

Departamento de Informática

PROYECTO FIN DE CARRERA
Ingeniería en Informática

SISTEMA DE GESTIÓN DE PERFILES POR UBICACIÓN PARA TELÉFONOS ANDROID

Autor: Pedro Palencia Herranz

Tutor: Javier Fernández Muñoz

Leganés, octubre de 2012

Agradecimientos

Quiero dar mi más sincero agradecimiento a Javier Fernández por haberme dirigido este proyecto. Por la libertad a la hora de elegir, las facilidades dadas y toda la ayuda prestada.

Quiero dar las gracias a mis compañeros de universidad y amigos Edu e Iñigo por hacer que estos largos y duros años de carrera hayan resultado más cortos y fáciles de lo que en realidad eran.

Gracias a mis amigos Juane, Charly, Marta, Rodri, y Ager por aguantarme todos estos años, por los buenos e inolvidables momentos que hemos vivido y por su apoyo incondicional.

Gracias también a toda mi familia por su confianza. En especial gracias a mis abuelos, mi tía Susana, mi tío Carlos y mis primos Carlos y Juan por todos los ánimos que me han dado a lo largo de la carrera.

Por último, quiero agradecer a mis padres, María José y Pedro, y mi hermana María José por su apoyo incondicional, por creer siempre en mí y animarme en los malos momentos. Gracias a mis padres por todo su trabajo y dedicación para darme una buena formación, sin ellos no hubiese sido posible nada de esto.

A todos muchas gracias.

Resumen

El objetivo de este proyecto es desarrollar un sistema para cambiar automáticamente la configuración del teléfono(WIFI, bluetooth, datos móviles y modo de sonido) dependiendo de la ubicación. El problema que este proyecto pretende subsanar es el de tener que modificar manualmente la configuración del teléfono cada vez que se llega a un lugar.

A la hora de desarrollar este proyecto se ha realizado un estudio minucioso de las tecnologías que se han utilizado, así como de las aplicaciones similares que existen. Se ha realizado también un estudio del sistema y de todas la funcionalidad necesaria que se requiere para cumplir todos los objetivos del proyecto.

En las próximas páginas se van a detallar las tecnologías utilizadas, así como los objetivos que se han marcado, seguido de un análisis y diseño del sistema. Por último se detalla la forma de implementar el sistema.

Para solucionar el problema se plantea crear una aplicación móvil que podrá ser utilizada por varios usuarios. Todos los datos que utiliza la aplicación se guardará en una base de datos SQLite. El usuario podrá crear nuevos perfiles, modificarlos y borrarlos. El sistema cambiará automáticamente de perfiles dependiendo de la localización.

La conclusión principal obtenida al realizar el proyecto es que se consigue encontrar una solución a un problema que afecta a muchos usuarios de dispositivos móviles.

Palabras clave: Android, perfil, ubicación, localización, aplicación móvil, Wi-Fi, células GSM, modo sonido, bluetooth, datos móviles.

Abstract

The main target of this project is to develop a system to automatically change the phone settings (Wi-Fi, bluetooth, mobile data and sound mode) depending on the location. The problem that this project tries to solve is the one that happens when the user has to manually change the phone settings every time he get to a place.

When developing this project, an extended study of the technologies used and similiar applications that exist has been made. Also, it has been made a study of the system and all necessary functionality required to meet the project objectives.

In the following pages there will be a detailed study of the used technologies and the goals that have been set, followed by an analysis and design of the system. Finally it describes how to implement the system.

To solve the problem is to create a mobile application that can be used by multiple users. All data used by the application is stored in a SQLite database. The user can create, edit and delete profiles. The system will automatically switch profiles depending on the location.

The main conclusion obtained from carrying out the project is that it has been found a solution to a problem that affects many mobile devices users.

Keywords: Android, profile, location, mobile application, Wi-Fi, GSM cells, sound mode, bluetooth, mobile data.

Índice general

1. Introducción y objetivos	11
1.1 Introducción	12
1.2 Motivación	13
1.3 Objetivos.....	14
1.4 Posibles aplicaciones del sistema	15
1.5 Contenido de la memoria	16
2. Estado del arte	17
2.1 Introducción	18
2.2 Android.....	19
2.2.1 Android vs iOS.....	19
2.2.2 Introducción a Android.....	20
2.2.3 Versiones de Android	22
2.2.4 Arquitectura de Android	25
2.3 SQLite	28
2.3.1 Introducción a SQLite.....	28
2.3.2 Características de SQLite.....	28
2.3.3 SQLite en Android.....	29
2.4 Aplicaciones similares	30
3. Análisis del sistema	31
3.1 Introducción	32
3.2 Descripción general	33
3.2.1 Capacidades generales	33
3.2.2 Restricciones generales	34
3.2.3 Características de los usuarios.....	34
3.2.4 Entorno operacional.....	34
3.3 Requisitos de usuario	35
3.3.1 Requisitos de capacidad	36
3.3.2 Requisitos de restricción	40
3.4 Casos de uso.....	42
3.4.1 Diagramas de casos de uso.....	42
3.4.2 Descripción textual	43
3.5.1 Requisitos de software funcionales	50
3.5.2 Requisitos de software no funcionales	55
3.6 Matrices de trazabilidad.....	60
3.6.1 Matriz de trazabilidad RUC - RSF	61
3.6.2 Matriz de trazabilidad RUR - RSNF	62
4. Diseño del sistema	63
4.1 Arquitectura del sistema	64
4.1.1 Modelo Vista Controlador (MVC)	64
4.2 Diseño detallado.....	66
4.2.1 Base de datos	66
4.2.2 Diseño de interfaces de usuario	68
4.2.2 Especificación de clases.....	70
4. Implementación y pruebas.....	78
5.1 Implementación	79
5.1.1 Creación de la base de datos	79
5.1.2 Creación de las pantallas Android	80

5.1.3 Creación de los servicios	83
5.2 Pruebas	85
5.2.1 Pruebas de aceptación	85
5.2.2 Matriz de trazabilidad	88
6. Conclusiones	91
5.1 Conclusiones	92
5.2 Líneas futuras	93
7. Acrónimos y abreviaturas	94
8. Bibliografía	97
Anexo A: Planificación y presupuesto	99
1. Planificación	100
2. Presupuesto	102
2.1 Coste de personal	102
2.2 Costes de hardware y software	102
2.3 Coste total	103
Anexo B: Manual de Usuario	104
1. Instalación	105
2. Uso de la aplicación	106

Índice de Ilustraciones

Ilustración 1: Android	19
Ilustración 2: Android vs iOS.....	19
Ilustración 3: Arquitectura de Android.....	21
Ilustración 4: Android 1.0.....	22
Ilustración 5: Android 1.1.....	22
Ilustración 6: Android 1.5.....	22
Ilustración 7: Android 1.6.....	23
Ilustración 8: Android 2.0.....	23
Ilustración 9: Android 2.2.....	23
Ilustración 10: Android 2.3.....	24
Ilustración 11: Android 3.0.....	24
Ilustración 12: Android 4.0.....	25
Ilustración 13: Android 4.1.....	25
Ilustración 14: Arquitectura Android	26
Ilustración 15: Logo SQLite.....	28
Ilustración 16: Casos de uso	42
Ilustración 17: Modelo Vista Controlador.....	64
Ilustración 18: Base de datos	66
Ilustración 19: Diagrama de navegación	70
Ilustración 20: Diagrama de clases	71
Ilustración 21: Pantalla de inicio	80
Ilustración 22: Pantalla de registro	81
Ilustración 23: Pantalla de perfiles	82
Ilustración 24: Pantalla de perfiles 2	82
Ilustración 25: Pantalla nuevo/modificar perfil.....	83
Ilustración 26: Planificación de tareas	100
Ilustración 27: Diagrama de Grantt.....	101
Ilustración 28: Pantalla de inicio de sesión	106
Ilustración 29: Pantalla de registro de usuario.....	107
Ilustración 30: Pantalla de perfiles	109
Ilustración 31: Pantalla de perfiles 2	109
Ilustración 32: Pantalla añadir/modifcar perfil.....	109
Ilustración 33: Pantalla de Log.....	110

Índice de Tablas

Tabla 1: Plantilla de requisito de usuario.....	35
Tabla 2: RUC-01.....	36
Tabla 3: RUC-02.....	36
Tabla 4: RUC-03.....	36
Tabla 5: RUC-04.....	37
Tabla 6: RUC-05.....	37
Tabla 7: RUC-06.....	37
Tabla 8: RUC-07.....	38
Tabla 9: RUC-08.....	38
Tabla 10: RUC-09.....	38
Tabla 11: RUC-10.....	38
Tabla 12: RUC-11.....	39
Tabla 13: RUC-12.....	39
Tabla 14: RUC-13.....	39
Tabla 15: RUC-14.....	39
Tabla 16: RUR-01.....	40
Tabla 17: RUR-02.....	40
Tabla 18: RUR-03.....	40
Tabla 19: RUR-04.....	40
Tabla 20: RUR-05.....	41
Tabla 21: RUR-06.....	41
Tabla 22: RUR-07.....	41
Tabla 23: RUR-08.....	41
Tabla 24: Plantilla de caso de uso.....	43
Tabla 25: CU-01.....	44
Tabla 26: CU-02.....	44
Tabla 27: CU-03.....	44
Tabla 28: CU-04.....	45
Tabla 29: CU-05.....	45
Tabla 30: CU-06.....	45
Tabla 31: CU-07.....	46
Tabla 32: CU-08.....	46
Tabla 33: CU-09.....	46
Tabla 34: CU-10.....	47
Tabla 35: CU-11.....	47
Tabla 36: CU-12.....	47
Tabla 37: CU-13.....	47
Tabla 38: CU-14.....	48
Tabla 39: Plantilla de requisito de software.....	49
Tabla 40: RSF-01.....	50
Tabla 41: RSF-02.....	50
Tabla 42: RSF-03.....	50
Tabla 43: RSF-04.....	51
Tabla 44: RSF-05.....	51
Tabla 45: RSF-06.....	52
Tabla 46: RSF-07.....	52

Tabla 47: RSF-08	52
Tabla 48: RSF-09	53
Tabla 49: RSF-10	53
Tabla 50: RSF-11	53
Tabla 51: RSF-12	53
Tabla 52: RSF-13	54
Tabla 53: RSF-14	54
Tabla 54: RSF-15	54
Tabla 55: RSF-16	54
Tabla 56: RSF-17	55
Tabla 57: RSF-18	55
Tabla 58: RSNF-01	55
Tabla 59: RSNF-02	56
Tabla 60: RSNF-03	56
Tabla 61: RSNF-04	56
Tabla 62: RSNF-05	56
Tabla 63: RSNF-06	57
Tabla 64: RSNF-07	57
Tabla 65: RSNF-08	57
Tabla 66: RSNF-09	58
Tabla 67: RSNF-10	58
Tabla 68: RSNF-11	58
Tabla 69: RSNF-12	58
Tabla 70: RSNF-13	59
Tabla 71: Matriz de trazabilidad RUC - RSF	61
Tabla 72: Matriz de trazabilidad RUC - RSF	62
Tabla 73: Plantilla de tabla de BBDD	66
Tabla 74: T-01	67
Tabla 75: T-02	67
Tabla 76: T-03	67
Tabla 77: T-04	68
Tabla 78: T-05	68
Tabla 79: T-06	68
Tabla 80: Plantilla de clase	72
Tabla 81: CL-01	72
Tabla 82: CL-02	73
Tabla 83: CL-03	73
Tabla 84: CL-04	73
Tabla 85: CL-05	74
Tabla 86: CL-06	74
Tabla 87: CL-07	75
Tabla 88: CL-08	75
Tabla 89: CL-09	75
Tabla 90: CL-10	76
Tabla 91: CL-11	77
Tabla 92: Plantilla de prueba de aceptación	85
Tabla 93: PR-01	85
Tabla 94: PR-02	85
Tabla 95: PR-03	85

Tabla 96: PR-04	86
Tabla 97: PR-05	86
Tabla 98: PR-06	86
Tabla 99: PR-07	86
Tabla 100: PR-08.....	86
Tabla 101: PR-09.....	86
Tabla 102: PR-10.....	86
Tabla 103: PR-11.....	86
Tabla 104: PR-12.....	86
Tabla 105: PR-13.....	87
Tabla 106: PR-14.....	87
Tabla 107: PR-15.....	87
Tabla 108: PR-16.....	87
Tabla 109: PR-17.....	87
Tabla 110: PR-18.....	87
Tabla 111: PR-19.....	87
Tabla 112: PR-20.....	87
Tabla 113: PR-21.....	87
Tabla 114: PR-22.....	88
Tabla 115: PR-23.....	88
Tabla 116: PR-24.....	88
Tabla 117: PR-25.....	88
Tabla 118: PR-26.....	88
Tabla 119: Matriz de trazabilidad PR-RSF	89
Tabla 120: Matriz de trazabilidad RSNF-PR.....	90
Tabla 121: Tabla coste de personal	102
Tabla 122: Tabla de coste de hardware y software.....	102
Tabla 123: Tabla de costes totales	103

1. Introducción y objetivos

1.1 Introducción

La finalidad de este capítulo es dar a conocer el contenido de este documento y que sirva como un primer encuentro con este proyecto final de carrera que se ha realizado. Este capítulo sirve como una pequeña introducción de los conceptos que se explican en los sucesivos apartados del documento.

El objetivo de este proyecto es resolver el problema de tener que cambiar la configuración del teléfono cada vez que llegamos a un sitio diferente. Lo que busca este proyecto es automatizar el cambio de la configuración basándose en la ubicación. Además se permitirá el uso de la aplicación por varios usuarios, pensando en dispositivos usados por diferentes personas.

En los últimos años ha habido un gran cambio en cuanto a las tecnologías móviles e internet. Estos cambios han generado nuevas necesidades en la sociedad. Un ejemplo sería la necesidad de poseer una conexión a internet de alta velocidad. En el mundo de las tecnologías móviles ha sucedido algo parecido. Ahora mismo gran parte de la población posee un teléfono inteligente. Han aparecido nuevos sistemas operativos, como es el caso de Android, y esto a llevado a crear nuevas necesidades en el uso de estos teléfonos inteligentes.

Un ejemplo de estas nuevas necesidades con respecto a los teléfonos es el de tener una conexión de datos móviles. Hoy en día el teléfono ya no sólo se utiliza para realizar llamadas y mandar mensajes. Ahora el teléfono tiene otras muchas posibilidades y muchas nuevas necesidades que los desarrolladores tratan de solucionar y cubrir.

Este proyecto fin de carrera ofrece una solución al problema de tener que modificar la configuración del teléfono dependiendo del lugar en el que nos encontremos. Se ofrece una aplicación Android que modifica automáticamente la configuración del teléfono dependiendo de la ubicación.

1.2 Motivación

Con la rápida evolución de la tecnología de los teléfonos móviles, que han dado lugar a los actuales teléfonos inteligentes, han ido surgiendo gran cantidad de nuevas necesidades, tal y como se indica en el apartado anterior.

Este proyecto fin de carrera se plantea como una solución al problema que se describirá detalladamente y con ejemplos a continuación: los nuevos teléfonos inteligentes tienen una gran cantidad de ajustes que en muchas ocasiones modificamos dependiendo de la ubicación y que a veces nos podemos olvidar cambiar la configuración, dando lugar a situaciones incómodas.

A continuación se detallan varios ejemplos de problemas que este proyecto soluciona:

1. En muchas ocasiones al llegar al lugar de trabajo cambiamos el modo de sonido del teléfono y lo ponemos en silencio o vibración para no molestar. Si se nos olvida realizar este cambio en la configuración, el teléfono puede sonar en cualquier momento. Este ejemplo serviría también para hospitales.
2. Con el fin de ahorrar batería, de corta duración en los nuevos teléfonos inteligentes, desactivamos los datos móviles al llegar a una zona con Wi-Fi, al llegar a casa por ejemplo. Como en el anterior caso, se nos puede olvidar realizar este cambio, y el teléfono se puede quedar sin batería.

Existen más ejemplos de estos nuevos problemas, y, gracias a la existencia de los nuevos sistemas operativos para dispositivos móviles podemos solucionarlos de una forma fácil.

Por lo tanto, la motivación principal de este proyecto es solucionar este nuevo problema aprovechando las posibilidades que nos ofrecen las plataformas móviles. El proyecto automatiza este cambio de configuración del teléfono.

1.3 Objetivos

El objetivo fundamental del proyecto es dar solución a los problemas anteriormente descritos. Este objetivo se define como:

- Desarrollar una aplicación que modifique la configuración del teléfono basándose en la ubicación y de forma automatizada, utilizando para ello diferentes perfiles.

A partir de este objetivo principal, se proponen los siguientes objetivos secundarios:

- Crear un sistema que automatice los cambios entre perfiles.
- Adquirir los conocimientos para poder implementar dicho sistema en una plataforma móvil.
- Diseñar una base de datos capaz de almacenar todos los datos necesarios para que el sistema funcione correctamente.
- Diseñar un sistema con una interfaz fácil e intuitiva y que no requiera un aprendizaje.
- Diseñar un prototipo funcional que se pueda ejecutar en un terminal real.

1.4 Posibles aplicaciones del sistema

Cualquier usuario de plataformas móviles se podrá beneficiar de este sistema. Además tiene otra ventaja, y es que al soportar varios usuarios, se podrá también utilizar en dispositivos compartidos.

El usuario se beneficiará del sistema ya que todo el proceso de modificar la configuración dependiendo de la ubicación queda totalmente automatizado. Al estar automatizado todo el proceso se le ahorrará tiempo.

El sistema se instalará en el teléfono y no hará falta ninguna modificación del teléfono para que el sistema comience a funcionar. Simplemente requerirá la creación de perfiles.

Los usuarios podrán registrarse y gestionar los perfiles que tienen creados, pudiendo crear tantos como quieran. El usuario podrá también ver el registro de los perfiles que se han ido activando, pudiendo controlar los sitios en los que ha estado.

1.5 Contenido de la memoria

En este apartado se va a proceder a explicar el contenido del documento, viendo cada uno de los capítulos que nos encontraremos. Se ofrecerá una breve explicación de cada uno de ellos.

Las secciones que componen este documento son las siguientes:

1. **Introducción y objetivos:** en esta apartado se mostrará la motivación del proyecto, así como los objetivos que se desean alcanzar centrándonos en los problemas que se pretenden resolver.
2. **Estado del arte:** en esta sección se describirá cada una de las tecnologías utilizadas. Principalmente se centrará en la plataforma Android sobre la que se desarrolla el proyecto, sin dejar de lado la base de datos SQLite que utilizará la aplicación. También se hará un análisis de las aplicaciones parecidas.
3. **Análisis del sistema:** en este apartado analizaremos de forma detallada las funcionalidades requeridas del sistema. Se especificarán los requisitos del sistema, tanto requisitos de usuario como requisitos software, y se analizarán también los diferentes casos de uso del sistema.
4. **Diseño del sistema:** en este apartado se realizará un estudio del sistema a implementar. Se definirá la arquitectura de dicho sistema, así como la especificación detallada de los diferentes componentes que lo conforman.
5. **Implementación:** en esta sección se podrán ver los diferentes detalles de cada uno de los módulos que contiene el sistema desde el punto de vista de la implementación. Se describirán las diferentes estrategias seguidas a la hora de desarrollar el sistema así como las dificultades surgidas.
6. **Conclusiones:** en este apartado se especificarán los problemas que se han ido encontrando en el desarrollo del proyecto, así como una breve conclusión del proyecto. También se explicarán los diferentes caminos que podrá tomar el proyecto, así como las mejoras propuestas para la aplicación.
7. **Referencias y abreviaturas:** se describirán cada una de las referencias y abreviaturas utilizadas a lo largo del documento.
8. **Bibliografía:** en este apartado se mostrará todo el material consultado a la hora de realizar el proyecto.

2. Estado del arte

2.1 Introducción

En este apartado se presenta un estudio de las diferentes tecnologías existentes hasta el momento con respecto al sistema a construir. El fin de este estudio es poder obtener un producto final de calidad.

Además en este capítulo se van a analizar aplicaciones similares que existen en el mercado comentando sus características, y realizando una comparativa entre ellas, analizando las diferentes ventajas que aportan cada una.

El objetivo final de este estudio es obtener una visión del entorno tecnológico y de otros productos similares, para poder evaluar si el producto a desarrollar es viable en la actualidad.

2.2 Android

Android es un sistema operativo móvil basado en el kernel de Linux. Está enfocado para ser utilizado en dispositivos móviles como teléfonos inteligentes, tabletas y otros dispositivos. Está desarrollado por la Open Handset Alliance, la cual es liderada por Google.



Ilustración 1: Android

Android permite el control de dispositivos a través de librerías desarrolladas o adaptadas por Google mediante el lenguaje de programación Java. Android es una plataforma de código abierto, es decir, que cualquier desarrollador puede crear y desarrollar aplicaciones.

2.2.1 Android vs iOS

A la hora de realizar el proyecto se han tomado una serie de decisiones con respecto a la plataforma en la que desarrollar el proyecto. Esta decisión se ha tomado basándose en las necesidades que se tenían en torno al software y hardware del proyecto.

Desde el momento inicial se pensó en dos plataformas móviles para el desarrollo del proyecto, las dos más importantes del mercado: iOS y Android. Se pensó siempre en



Ilustración 2: Android vs iOS

estas dos plataformas por su gran madurez e integración en los mercados. Debido a estas dos razones se descartaron otras como son Windows Phone 7, Bada, WebOS, RIM o Symbian. Otra razón por la que desde el principio se pensó en estas dos plataformas móviles es porque son muy atractivas para el desarrollador: poseen un número muy alto de usuarios.

Otras características que hace a iOS y Android tan atractivos es que son dispositivos con interfaces táctiles y con el hardware más avanzado (teléfonos con hasta 4 núcleos).

El único problema de iOS es que solo se utiliza en dispositivos Apple, mientras que Android lo utilizan un gran número de marcas, tales como Samsung, HTC, LG o Sony, entre otras muchas. Esto es a su vez un punto a favor de iOS con respecto a Android, ya que a la hora de desarrollar una aplicación en iOS sólo se tiene un dispositivo, mientras que en Android hay un gran número de dispositivos, con diferentes características. Una de las características más importantes a la hora de desarrollar una aplicación es el tamaño de la pantalla: mientras que en Apple tenemos un tamaño fijo, en Android existe una gran cantidad de dimensiones, así como de resoluciones, por lo tanto hay que tener este aspecto en cuenta a la hora de desarrollar una aplicación.

El motivo principal para haber elegido Android como plataforma final para desarrollar la aplicación es que para implementar en Android se utiliza el lenguaje de programación Java y para iOS se utiliza objective C. Ya que de Java tengo un amplio conocimiento y experiencia, mientras que de objective C no tengo ningún tipo de experiencia, y requeriría un periodo de aprendizaje, la elección de la plataforma será Android. También influyó la dificultad que existe en iOS para modificar los parámetros del Wi-Fi, bluetooth, etc. debido a la seguridad sandbox de las aplicaciones desarrolladas para iOS.

Una vez elegida Android como plataforma para desarrollar el proyecto, se pasa a elegir el dispositivo móvil que se utilizará para desarrollar la aplicación. Para este proyecto da igual uno que otro, ya que no tiene restricciones de hardware importantes, pero, debido a la disponibilidad, se ha elegido un Sony Xperia U, con una versión de Android 2.3.

2.2.2 Introducción a Android

Android es un sistema operativo móvil basado en Linux, que junto con aplicaciones middleware está enfocado a ser utilizado en dispositivos móviles como teléfonos inteligentes, tabletas, Google TV y otros dispositivos, tales como relojes, reproductores musicales, ordenadores, Netbook e incluso televisores. Es desarrollado por la Open Handse Alliance, que está liderada por Google.

El sistema Android fue desarrollado inicialmente por Android Inc., que fue adquirida por Google en el año 2005. Pero hasta el año 2007, con la formación de la compañía Open Handset Alliance, no se estrenó. La Open Handset Alliance es un consorcio de 78 compañías de hardware software y telecomunicaciones entre las que se encuentran Texas Instruments, Broadcom Corporation, Nvidia, Qualcomm, Samsung Electronics, Sprint Nextel, Intel, LG, Motorola y T-Mobile. El objetivo de este consorcio de empresas era el de desarrollar estándares abiertos para dispositivos móviles.

La estructura del sistema operativo Android se compone de aplicaciones que se ejecutan en un framework Java de aplicaciones orientadas a objetos sobre el núcleo de las bibliotecas de Java en una máquina virtual Dalvik con compilación en tiempo de ejecución. Las bibliotecas están escritas en lenguaje C y se compilan a código nativo de ARM. El responsable de la mayor cantidad de código fuente de Android bajo licencia de Software Apache ha sido Google, haciendo uso de una licencia de software libre y de código abierto a cualquier desarrollador.

Algunas de las bibliotecas escritas en lenguaje C son: interfaz gráfica, OpenGL, motor WebKit, base de datos SQLite, SSL, SGL, FreeType...

Las capas de las que se compone el sistema Android se pueden ver en la ilustración 3: aplicación, framework de aplicaciones, Android Runtime, librerías y el kernel de Linux. En el apartado 2.2.4 se explicarán estas capas más detalladamente.



Ilustración 3: Arquitectura de Android

Las principales características de Android son:

- Plataforma adaptable a pantallas de mayor resolución.
- Utilización de base de datos relacional SQLite.
- Soporte de un gran número de tecnologías de conectividad: GSM, EDGE, Wi-Fi, HSDPA, LTE...
- Soporte para SMS y MMS.
- Navegador web basado en WebKit con JavaScript.
- Soporte para gran número de formatos multimedia.
- Soporte para streaming.
- Soporte para hardware adicional: cámara de fotos, vídeos, GPS, acelerómetros, sensores de luz, de proximidad, etc.
- Posee un entorno de desarrollo, con emulador de dispositivos, herramienta de depuración de memoria, etc. El entorno de desarrollo es Eclipse.
- Tienda de aplicaciones Google Play.
- Soporte para pantallas multi-táctil.
- Soporte para bluetooth.
- Soporte para videollamada.
- Soporte para multitarea.
- Reconocimiento de voz.
- Soporte para tethering.

2.2.3 Versiones de Android

A largo del tiempo, Android ha ido desarrollando diferentes versiones, introduciendo nuevas mejoras en todas ellas, y cubriendo las nuevas necesidades de los diferentes dispositivos del mercado. En este apartado se van a explicar brevemente cada una de estas versiones, analizando las mejoras introducidas en cada una de ellas.

Android 1.0: Apple Pie



Ilustración 4:
Android 1.0

Primera versión comercial del Android, liberada el 23 de septiembre de 2008. Utilizada por el teléfono HTC Dream.

- Aplicaciones Google: Gmail, Google Calendar, Google Maps, Google Talk, etc.
- Navegador de internet con zoom.
- Aplicación Android Market para descargar y actualizar aplicaciones.
- Soporte para cámara.
- Acceso a email (POP3, IMAP4 y SMTP).
- Soporte para SMS y MMS.
- Reproductor multimedia.
- Soporte para notificaciones.
- Soporte para WIFI y Bluetooth.

Android 1.1: Banana Bread



Ilustración 5:
Android 1.1

Liberada el 9 de febrero de 2009.

- Añadidos detalles de negocios en Google Maps.
- Soporte para añadir archivos adjuntos a los mensajes.
- Posibilidad de mostrar el teclado durante las llamadas.

Android 1.5: Cupcake



Ilustración 6:
Android 1.5

Liberada el 30 de abril de 2009.

- Soporte para teclados con predicción de texto y diccionario.
- Soporte para widgets.
- Grabación y reproducción de videos.
- Soporte para manos libres bluetooth.
- Transacciones de pantalla animadas.
- Capacidad de subir fotos y videos.

- Añadida auto-rotación.

Android 1.6: Donut



Ilustración 7:
Android 1.6

Liberada el 15 de septiembre de 2009.

- La búsqueda muestra resultados del historial web y contactos.
- Los desarrolladores pueden añadir contenido a los resultados de búsquedas.
- Mejoras en la integración de la galería y la cámara, así como mejoras en la velocidad de acceso a la cámara.
- Posibilidad de borrar varias fotos a la vez.
- Actualización de soporte para CDMA/EVDO, 802.1x, VPNs y lectura de textos.
- Soporte para resoluciones de pantallas WVGA.
- Framework de gestos y herramienta de desarrollo GestureBuilder.

Android 2.0/2.1: Éclair



Ilustración 8: Android 2.0

Liberada el 26 de octubre de 2009.

- Posibilidad de añadir varias cuentas par sincronizar email y contactos.
- Bluetooth 2.1.
- Posibilidad de pulsar la foto de un contacto y seleccionar llamar, mandar SMS o mandar email al contacto.
- Posibilidad de buscar mensajes.
- Nuevas funciones para la cámara: soporte para flash, zoom digital, modos...
- Mejoras en el teclado.
- Velocidad del hardware optimizada.
- Soporte para más tamaños de pantalla y resoluciones.

Android 2.2: Froyo



Ilustración 9:
Android 2.2

Liberado el 20 de mayo de 2010.

- Optimización del sistema (velocidad, memoria y rendimiento).
- Integración del motor Javascript de Chrome.
- Mejorado el soporte de Microsoft Exchange.
- Funcionalidad de Wi-Fi hotspot y tethering por USB.
- Permite desactivar el tráfico de datos.
- Soporte para cambiar rápidamente de idioma en el teclado.

- Marcación por voz a través de bluetooth.
- Soporte para instalar aplicaciones en la memoria expandible.
- Soporte para Adobe Flash.
- Soporte para pantallas con un número alto de puntos por pulgadas, tales como pantallas de 4" 720p.

Android 2.3 Gingerbread



Liberado el 6 de diciembre de 2010.

Ilustración 10:
Android 2.3

- Mejora de la interfaz de usuario.
- Soporte para pantallas de gran tamaño y resolución.
- Mejoras en el teclado.
- Mejoras en el sistema de copiar/pegar.
- Soporte para Near Field Communication.
- Soporte nativo para telefonía VoIP SIP.
- Nuevos efectos de audio.
- Nuevo gestor de descargas.
- Soporte para varias cámaras.
- Soporte para reproducción de videos WebM/VP8 y decodificación de audio AAC.
- Mejora en la administración de energía y control de aplicaciones mediante un administrador de tareas.
- Cambio del sistema de archivos de YAFFS a ext4.
- Soporte nativo para nuevos sensores (giroscopios y barómetros).
- Soporte para video chat utilizando Google Talk.

Android 3.0/3.1/3.2: Honeycomb



Liberado el 22 de febrero de 2011.

Ilustración 11:
Android 3.0

- Enfocado a tabletas.
- Añadida la barra del sistema.
- Añadida la barra de acción.
- Sistema multitarea simplificado y mejorado.
- Teclado rediseñado.
- Mejorado el sistema de copiar/pegar.
- Mejorado el navegador web con ventanas, relleno automático y modo incognito.
- Acceso rápido a diferentes funciones de la cámara.
- Soporte para video chat usando Google Talk.
- Aceleración hardware.
- Soporte para procesadores multi-core.
- Soporte para periféricos y accesorios USB.

Android 4.0: Ice Cream Sandwich



Liberado el 19 de Octubre de 2011.

Ilustración 12:
Android 4.0

- Unifica el uso en cualquier dispositivo, tanto teléfonos, tabletas, televisores, etc.
- Posibilidad de agrupar aplicaciones en carpetas.
- Aceleración por hardware de la interfaz de usuario.
- Desbloqueo mediante reconocimiento facial.
- Nueva interfaz de usuario: Roboto.
- Gestor de tráfico de datos (avisos y alarmas cuando se exceden límites).
- Mejora de la aplicación de la cámara.
- Editor de fotos.
- Android Beam: permite compartir contenido entre teléfonos utilizando NFC.
- Grabación de video a 1080p.
- Soporte nativo para MKV.
- Soporte nativo para el uso de stylus.

Android 4.1: Jelly Bean



Liberado el 9 de julio de 2012.

Ilustración 13:
Android 4.1

- Mejora de la interfaz de usuario (suavizada).
- Posibilidad de apagar las notificaciones de una aplicación.
- Android Beam permite compartir archivos utilizando bluetooth.
- Dictado por voz sin conexión a internet.
- Mejora en la cámara.
- Mejora en el dictado por voz.
- Google Wallet.
- Fotos de contactos de alta resolución.

2.2.4 Arquitectura de Android

El sistema operativo Android está formado por varias capas que facilitan al desarrollador la creación de aplicaciones. Como hemos comentado anteriormente, Android utiliza el kernel modificado de Linux v2.6. En la siguiente imagen, tomada del sitio oficial de desarrolladores Android se puede apreciar la arquitectura de Android.

A continuación vamos a comentar cada una de las capas:

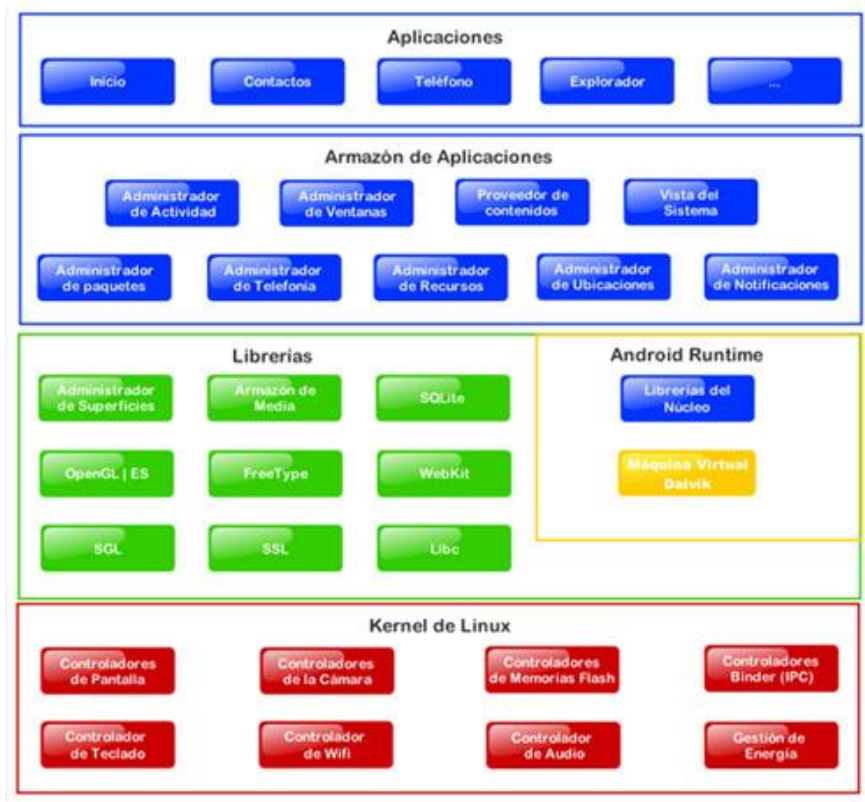


Ilustración 14: Arquitectura Android

Kernel de Linux

Esta capa contiene el kernel de Linux versión 2.6, similar al que se puede incluir en cualquier distribución de Linux, tal como Ubuntu, Fedora, Debian..., pero adaptado a las características del hardware en el que se ejecutará Android, es decir, dispositivos móviles.

Esta capa actúa como una capa de abstracción entre el hardware del dispositivo (pantalla, cámara, bluetooth, etc.) y el resto de las capas de la arquitectura. El desarrollador no accede directamente a esta capa, sino que utiliza las librerías disponibles en capas superiores. También evitamos de esta forma la necesidad de tener que conocer las características de los diferentes dispositivos hardware.

El kernel también se encarga de gestionar los recursos del dispositivo y del sistema operativo.

Librerías

Esta capa se sitúa justo encima del kernel y la componen las bibliotecas nativas de Android. Estas bibliotecas están escritas en C o C++ y compiladas para la arquitectura hardware específica del dispositivo. Estas bibliotecas están desarrolladas normalmente por el fabricante, que también se encarga de instalarlas en el dispositivo. El objetivo de

estas librerías es proporcionar funcionalidad a las aplicaciones para las tareas que se ejecutan con frecuencia, evitando tener que codificarlas cada vez y garantizando que se llevan a cabo de la forma más eficiente.

Entre las librerías incluidas suelen estar OpenGL, bibliotecas multimedia, Webkit, SSL, FreeType, SQLite...

Entorno de ejecución

Tal y como se puede ver en la ilustración 14, el entorno de ejecución de Android no es una capa en sí mismo. Aquí se encuentran las librerías con las funcionalidades habituales de Java así como otras de Android.

En componente principal es la máquina virtual Dalvik. Esta máquina virtual está optimizada para requerir poca memoria y está diseñada para ejecutar varias instancias de la máquina virtual de manera simultánea.

Las aplicaciones se codifican en Java y se compilan en un formato específico para esta máquina virtual. De esta forma, las aplicaciones sólo se compilan una vez, y están listas para ejecutarse en cualquier dispositivo Android. Cada aplicación utiliza su propia instancia de máquina virtual Dalvik.

Marco de aplicaciones

Está formado por todas las clases y servicios que utilizan directamente las aplicaciones para realizar sus funciones. La mayoría de los componentes de esta capa son librerías Java que acceden a las capas inferiores a través de la máquina virtual Dalvik. La arquitectura está diseñada para simplificar la reutilización de componentes.

Dentro de esta capa encontramos las siguientes aplicaciones: administrador de actividad, administrador de ventanas, proveedor de contenidos, vista del sistema, administrador de paquetes, administrador de telefonía, administrador de recursos, administrador de ubicaciones, administrador de notificaciones, administración de sensores y cámara.

Aplicaciones

Es la capa superior, en la que se incluyen todas las aplicaciones que contiene el dispositivo, tanto las que tienen interfaz de usuario como las que no, nativas (programadas en C o C++) y las administradas (programadas en Java), las que vienen preinstaladas en el dispositivo y aquellas instaladas por el usuario.

En esta capa se encuentran también las aplicaciones principales del sistema, como son los launchers o los widgets.

2.3 SQLite

Una base de datos es un conjunto de datos pertenecientes a un mismo contexto y almacenados sistemáticamente para su posterior uso. Existen programas denominados sistemas gestores de bases de datos que permiten almacenar y posteriormente acceder a los datos de forma rápida y estructurada.

SQLite es un sistema de gestión de bases de datos relacional. SQLite es un proyecto de dominio público creado por D. Richard Hipp. A diferencia de otros sistemas gestores de bases de datos cliente-servidor, SQLite se enlaza con el programa, pasando a ser parte integral del mismo.

2.3.1 Introducción a SQLite

SQLite es un sistema gestor de bases de datos relacional compatible con ACID, contenida en una pequeña biblioteca escrita en C.

A diferencia de los sistemas de gestión de bases de datos cliente-servidor, el motor de SQLite no es un proceso independiente con el que el programa principal se comunica.



Ilustración 15: Logo SQLite

En lugar de eso, la biblioteca SQLite se enlaza con el programa, pasando a ser parte integral del mismo. El programa utiliza la funcionalidad de SQLite a través de llamadas simples a subrutinas y funciones. Esto reduce la latencia en el acceso a la base de datos, ya que las llamadas son más eficientes que la comunicación entre procesos de las bases de datos cliente-servidor. El conjunto de la base de datos son

guardados como un único fichero en la máquina. Para que esto funcione correctamente, se bloquea todo el fichero al principio de cada transición.

Existen muchos programas que utilizan bases de datos SQLite. Entre ellos están: Adobe Photoshop Elements, Mozilla Firefox, Opera, Skype... Además está incluido en numerosos sistemas operativos: Android, Blackberry, iOS, Maemo, MeeGo, Symbian, webOS...

2.3.2 Características de SQLite

La biblioteca de SQLite implementa casi por completo el estándar SQL-92. Incluye transacciones de bases de datos atómicas, consistencia de base de datos, aislamiento y durabilidad, triggers y la mayor parte de las consultas complejas.

El modo en el que SQLite trabaja con los diferentes tipos de datos es inusual. Asigna el tipo de datos a los valores individuales, en vez de a la columna, como la mayoría de las bases de datos SQL. Gracias a esto se puede insertar un String en una columna numérica.

Varios procesos pueden acceder a la base de datos: se pueden producir varios accesos de lectura, pero solo se puede producir un acceso de escritura si no existe ningún otro acceso concurrentemente.

2.3.3 SQLite en Android

Android ofrece soporte completo para bases de datos SQLite. Las bases de datos serán accesibles por cualquier clase de la aplicación, pero será inaccesible desde fuera de la aplicación. La base de datos SQLite en Android no necesita ningún tipo de administración.

El método recomendado para crear una base de datos en Android, según la web de desarrolladores Android, es crear una subclase de SQLiteOpenHelper e implementar el método onCreate().

A partir de aquí, Android aporta todo lo necesario para interactuar con la base de datos. Lo único que hay que hacer es acceder a la base de datos y utilizar las funciones que la API de Android proporciona.

Directamente se pueden hacer consultas, actualizar, insertar datos, borrar, e incluso se pueden ejecutar secuencias en lenguaje SQLite.

2.4 Aplicaciones similares

Existen varias aplicaciones similares a la que buscamos desarrollar. Estas aplicaciones son: Llama, desarrollada por KebabApps y gratuita, y Locale, desarrollada por Two Forty Four a.m. LLC, con un precio de 4,99€.

Llama basa su actividad en tres tipos diferentes de información. Por un lado están las diferentes áreas. Las áreas se aprenden utilizando las torres de telefonía, guardando los identificadores de las células GSM. Por otro lado tenemos los eventos, utilizados para definir acciones que se realizarán al producirse el evento. Dentro de las acciones que puede realizar se encuentran: apagar datos móviles, encender bluetooth, apagar Wi-Fi, etc. Existen también multitud de condiciones a partir de las cuales se ejecutan las acciones, como las horas, llegada a ubicación, conectarse a una red Wi-Fi, etc. Por último están los perfiles, en los que se definen los diferentes volúmenes: llamadas, mensajes, notificaciones, etc. Utilizando todo esto modifica toda la configuración del teléfono basándose en la ubicación.

Esta aplicación requiere de un periodo de aprendizaje ya que es bastante compleja. Requiere además un largo periodo de puesta a punto, para que funcione como se espera. Por otro lado, la elección de la forma de aprender la ubicación, utilizando las células GSM parece que pueda llegar a ser un poco imprecisa.

Por otro lado está la aplicación Locale. Locale utiliza lo que ellos llaman situaciones. Estas situaciones funcionan basándose en condiciones y ajustes. Dentro de las condiciones nos encontramos con la localización, la hora, el nivel de batería, e incluso con la orientación del teléfono. Cuando estas condiciones se cumplen, el teléfono cambia los ajustes utilizando los que tiene guardados para esa situación. Con Locale se pueden ajustar el bluetooth, el volumen, el Wi-Fi, el fondo de pantalla, etc.

Al igual que la aplicación Llama, el periodo de puesta a punto es demasiado extenso, ya que tiene demasiadas opciones.

La conclusión sobre estas aplicaciones es que son demasiado complejas para la mayoría de los usuarios de teléfonos inteligentes. El tiempo de aprendizaje es demasiado extenso, y, en muchos casos, no son intuitivas. Además la aplicación Locale tiene un precio que mucha gente no está dispuesta a pagar.

3. Análisis del sistema

3.1 Introducción

En este apartado se va a definir el análisis del sistema a desarrollar. Este apartado se basa en los estándares recogidos por la Agencia Espacial Europea en su versión Lite para proyectos de Ingeniería de Software. Se ha escogido la versión Lite de la ESA ya que está diseñada para proyectos más pequeños, como es este que se desarrolla. Se ha modificado ligeramente algunos apartados, ya que después de analizarlos, se ha visto que no son necesarios.

Una vez establecidas las tecnologías que se utilizarán para el desarrollo de la aplicación, se van a definir todas las funcionalidades requeridas por el sistema. Para ello, primero se proporcionará una visión general del mismo. Se analizarán las capacidades y restricciones del sistema.

A continuación se llevará a cabo la definición de los requisitos de usuario, nacidos a partir de los primeros requisitos y que son una especificación completamente detallada de la funcionalidad que poseerá el sistema desarrollado. A partir de estos requisitos de usuario se definirán los casos de uso, y, por último, y a partir de todo lo anterior, se definirán los requisitos software.

3.2 Descripción general

Con este punto se quiere dar una visión más detallada del sistema a desarrollar, de una forma lo más sencilla posible para que se pueda entender el producto que se va a desarrollar.

Se irán detallando por puntos los distintos aspectos del sistema, introduciéndose cada uno de ellos en una parte del desarrollo. Con esto se intenta que se puedan comprender los distintos aspectos a tratar durante el desarrollo del sistema.

3.2.1 Capacidades generales

En este apartado se define cuál es el objetivo que debe cumplir el sistema de cara a los usuarios. Para ello, a continuación se definen las capacidades que debe de cumplir el sistema y que en sucesivos apartados se detallarán.

- El producto a desarrollar consiste en una aplicación que modifique la configuración del teléfono basándose en la ubicación, utilizando para ello diferentes perfiles.
- Cualquier usuario se podrá registrar en el sistema.
- Cualquier usuario registrado podrá iniciar sesión en la aplicación.
- Se podrán crear nuevos perfiles.
- Se podrán modificar cualquiera de los perfiles.
- Se podrá aprender la ubicación de cualquiera de los perfiles, excepto del perfil por defecto.
- Se podrá borrar la ubicación de los perfiles.
- Se podrá cancelar el aprendizaje de una ubicación en curso.
- Se podrá borrar cualquiera de los perfiles, excepto del perfil por defecto.
- Los ajustes que se podrán definir dependiendo de los perfiles son: Wi-Fi, bluetooth, datos móviles y modo de sonido.
- Cualquier usuario que ha iniciado sesión podrá borrar su cuenta.
- Cualquier usuario que ha iniciado sesión podrá ver un registro de los perfiles que se han ido activando a lo largo del tiempo.

3.2.2 Restricciones generales

Para que la aplicación funcione correctamente han de respetarse una serie de restricciones. En este apartado se comentan las restricciones que tendrá el sistema y que son necesarias para diseñarlo.

- Las personas que usarán la aplicación no tienen por qué tener amplios conocimientos en el uso de aplicaciones tecnológicas o informáticas, por lo que se pondrá especial interés en que la forma de usar el sistema sea lo más sencilla posible y se presente al usuario mediante una interfaz intuitiva.
- Una vez haya un usuario que ha iniciado sesión, el sistema detectará cada un periodo de tiempo la ubicación, y cambiará los ajustes dependiendo del perfil activo. Para determinar la ubicación se utilizarán las células GSM, el router Wi-Fi al que está conectado y la localización (latitud más longitud).
- Toda la información almacenada por la aplicación se guardará en una base de datos SQLite.
- La aplicación será mostrada al usuario en castellano.

3.2.3 Características de los usuarios

El sistema tendrá un único tipo de usuario. Este usuario podrá, a través de la aplicación, gestionar los perfiles. Podrá crear nuevos perfiles, modificarlos y borrarlos. Podrá también hacer que el sistema aprenda la ubicación para los diferentes perfiles.

Por otro lado, no es necesario que los usuarios de la aplicación tengan elevados conocimientos informáticos para utilizar el sistema ya que tendrá una interfaz usable e intuitiva.

3.2.4 Entorno operacional

En este apartado se analiza la tecnología del sistema.

- Se requerirá un dispositivo Android con la versión 2.2 Froyo como mínimo.
- Se requiere tener instalada la aplicación en este dispositivo.
- Para el correcto funcionamiento del sistema es recomendable tener activada la ubicación por Wi-Fi.
- Para la base de datos que almacenará todos los datos de la aplicación se utilizará una base de datos SQLite.

3.3 Requisitos de usuario

El objetivo de este apartado es detallar todos los requisitos de usuario que el sistema necesitará para cumplir todas las funcionalidades necesarias. Estos requisitos de usuarios servirán para diseñar en futuros apartados el sistema de gestión de perfiles por ubicación.

Estos requisitos se dividen en requisitos de capacidad y de restricción. Los primeros indican qué debe ser capaz de realizar el sistema y los segundos especifican las restricciones que establecen el cómo se realizan las tareas que fueron indicadas en los primeros.

El formato que se utilizará para recoger los requisitos de usuario es el siguiente:

Identificador	
Título	
Prioridad	<input type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Necesidad	<input type="checkbox"/> Esencial <input type="checkbox"/> Conveniente <input type="checkbox"/> Opcional
Fuente	
Descripción	

Tabla 1: Plantilla de requisito de usuario

Dónde cada campo representa lo siguiente:

- **Identificador:** código alfanumérico único que identifica a cada requisito. Comienza por RU seguido de una C si es de capacidad o una R si es de restricción. Para concluir llevan un número identificador del requisito.
- **Título:** es el nombre único del requisito.
- **Prioridad:** hace referencia al orden temporal en que debe realizarse el requisito.
- **Necesidad:** establece la importancia del requisito desde el punto de vista del cliente.
- **Fuente:** indica la procedencia del requisito.
- **Descripción:** detalla la especificación del requisito de manera precisa y simple.

3.3.1 Requisitos de capacidad

Los requisitos del usuario de capacidad definen las funciones y operaciones requeridas por los usuarios para resolver un problema o alcanzar un objetivo. Describen una operación o secuencia de operaciones que el software debe ser capaz de realizar.

Identificador	RUC-01
Título	Registro de usuario
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Conveniente <input type="checkbox"/> Opcional
Fuente	Cliente
Descripción	
Permitir a cualquier usuario registrarse en el sistema. Para ello debe rellenar el formulario de registro, con los campos nombre de usuario y contraseña.	

Tabla 2: RUC-01

Identificador	RUC-02
Título	Inicio de sesión
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Conveniente <input type="checkbox"/> Opcional
Fuente	Cliente
Descripción	
Permitir iniciar sesión en la aplicación a cualquier usuario registrado. Para ello debe aportar su nombre de usuario y contraseña.	

Tabla 3: RUC-02

Identificador	RUC-03
Título	Perfil "Default"
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Conveniente <input type="checkbox"/> Opcional
Fuente	Cliente
Descripción	
Para cada cuenta de usuario existirá un perfil llamado "Default" que se activará cuando no haya ningún otro perfil activo.	

Tabla 4: RUC-03

Identificador	RUC-04
Título	Nuevo perfil
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Conveniente <input type="checkbox"/> Opcional
Fuente	Cliente
Descripción	
Permitir a cualquier usuario registrado y que ha iniciado sesión crear un nuevo perfil. Para crear un nuevo perfil debe de rellenar el formulario correspondiente, con los campos: <ul style="list-style-type: none">- Nombre.- Wi-Fi.- Bluetooth.- Datos Móviles.- Modo sonido.- Tiempo aprendizaje: si quiere aprender la ubicación en ese momento.	

Tabla 5: RUC-04

Identificador	RUC-05
Título	Modificar perfil
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Conveniente <input type="checkbox"/> Opcional
Fuente	Cliente
Descripción	
Permitir a cualquier usuario registrado y que ha iniciado sesión modificar un perfil. Los datos a modificar son: <ul style="list-style-type: none">- Nombre.- Wi-Fi.- Bluetooth.- Datos Móviles.- Modo sonido.	

Tabla 6: RUC-05

Identificador	RUC-06
Título	Borrar usuario
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Conveniente <input type="checkbox"/> Opcional
Fuente	Cliente
Descripción	
Permitir a cualquier usuario registrado y que ha iniciado sesión borrar su usuario.	

Tabla 7: RUC-06

Identificador	RUC-07
Título	Aprender ubicación perfil
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Conveniente <input type="checkbox"/> Opcional
Fuente	Cliente
Descripción	
Permitir a cualquier usuario registrado y que ha iniciado sesión aprender o volver a aprender la ubicación de un perfil.	

Tabla 8: RUC-07

Identificador	RUC-08
Título	Cancelar aprendizaje ubicación
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Conveniente <input type="checkbox"/> Opcional
Fuente	Cliente
Descripción	
Permitir a cualquier usuario registrado y que ha iniciado sesión cancelar el aprendizaje de la ubicación de un perfil siempre y cuando se esté realizando en ese momento.	

Tabla 9: RUC-08

Identificador	RUC-09
Título	Borrar ubicación perfil
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Conveniente <input type="checkbox"/> Opcional
Fuente	Cliente
Descripción	
Permitir a cualquier usuario registrado y que ha iniciado sesión borrar las ubicaciones guardadas de un perfil.	

Tabla 10: RUC-09

Identificador	RUC-10
Título	Borrar perfil
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Conveniente <input type="checkbox"/> Opcional
Fuente	Cliente
Descripción	
Permitir a cualquier usuario registrado y que ha iniciado sesión borrar un perfil.	

Tabla 11: RUC-10

Identificador	RUC-11
Título	Visualizar perfiles
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Conveniente <input type="checkbox"/> Opcional
Fuente	Cliente
Descripción	
Permitir a cualquier usuario registrado y que ha iniciado sesión ver los perfiles creados. En esta lista se mostrará el perfil activo y si hay algún perfil aprendiendo ubicación.	

Tabla 12: RUC-11

Identificador	RUC-12
Título	Visualizar log
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Conveniente <input type="checkbox"/> Opcional
Fuente	Cliente
Descripción	
Permitir a cualquier usuario registrado y que ha iniciado sesión ver el log de los perfiles que se han ido activando, con la fecha y hora en la que se han activado.	

Tabla 13: RUC-12

Identificador	RUC-13
Título	Cerrar sesión
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Conveniente <input type="checkbox"/> Opcional
Fuente	Cliente
Descripción	
Permitir a cualquier usuario registrado y que ha iniciado sesión cerrar su sesión.	

Tabla 14: RUC-13

Identificador	RUC-14
Título	Activar perfil
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Conveniente <input type="checkbox"/> Opcional
Fuente	Cliente
Descripción	
Se irán activando los perfiles dependiendo de la ubicación.	

Tabla 15: RUC-14

3.3.2 Requisitos de restricción

Los requisitos de usuario de restricción definen las restricciones impuestas por los usuarios sobre la manera en que el problema es resuelto o el objetivo es alcanzado. Restringen la manera en que el software es construido o funciona, sin alterar o describir las capacidades del software.

Identificador	RUR-01
Título	Idioma de la aplicación
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Conveniente <input type="checkbox"/> Opcional
Fuente	Cliente
Descripción	
El idioma del sistema será el castellano.	

Tabla 16: RUR-01

Identificador	RUR-02
Título	Funcionamiento en dispositivos Android
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Conveniente <input type="checkbox"/> Opcional
Fuente	Cliente
Descripción	
La aplicación funcionará únicamente en dispositivos con el sistema operativo Android con la versión 2.2 como mínimo.	

Tabla 17: RUR-02

Identificador	RUR-03
Título	Almacenaje de los datos
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Conveniente <input type="checkbox"/> Opcional
Fuente	Cliente
Descripción	
La aplicación almacenará todos los datos en una base de datos SQLite dentro de la aplicación Android.	

Tabla 18: RUR-03

Identificador	RUR-04
Título	Aprendizajes simultáneos
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Conveniente <input type="checkbox"/> Opcional
Fuente	Cliente
Descripción	
La aplicación no permitirá realizar más de un aprendizaje de ubicación simultáneamente.	

Tabla 19: RUR-04

Identificador	RUR-05
Título	Sesión persistente
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Conveniente <input type="checkbox"/> Opcional
Fuente	Cliente
Descripción	
Se recordará el inicio de sesión para evitar al usuario tener que introducirlos cada vez que inicie la aplicación.	

Tabla 20: RUR-05

Identificador	RUR-06
Título	Disponibilidad de Wi-Fi
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Conveniente <input type="checkbox"/> Opcional
Fuente	Cliente
Descripción	
Para el correcto funcionamiento de la aplicación es necesario disponer de un chip Wi-Fi.	

Tabla 21: RUR-06

Identificador	RUR-07
Título	Ubicación por Wi-Fi
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Conveniente <input type="checkbox"/> Opcional
Fuente	Cliente
Descripción	
Para el correcto funcionamiento de la aplicación es necesario tener activada la ubicación por Wi-Fi.	

Tabla 22: RUR-07

Identificador	RUR-08
Título	Determinar localización
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Conveniente <input type="checkbox"/> Opcional
Fuente	Cliente
Descripción	
Para determinar la localización, el sistema utilizará la red Wi-Fi y las células GSM.	

Tabla 23: RUR-08

3.4 Casos de uso

Los casos de uso son la representación gráfica de los requisitos, y por lo tanto, nacen de éstos. Un caso de uso representa una secuencia de interacciones que se desarrollará entre un sistema y sus actores en respuesta a un evento que inicia el actor principal sobre el propio sistema. El objetivo de esta apartado es representar y detallar las funcionalidades y tareas que puede realizar un usuario en el sistema, partiendo de la base de que el usuario controla el sistema mediante el dispositivo móvil que se ha especificado en los requisitos de usuario.

Los casos de uso son una técnica muy utilizada en el área de la ingeniería de software para mostrar la futura capacidad del sistema de una forma simple e intuitiva.

3.4.1 Diagramas de casos de uso

Los diagramas de casos de uso sirven para especificar la comunicación y el comportamiento de un sistema mediante su interacción con el sistema.

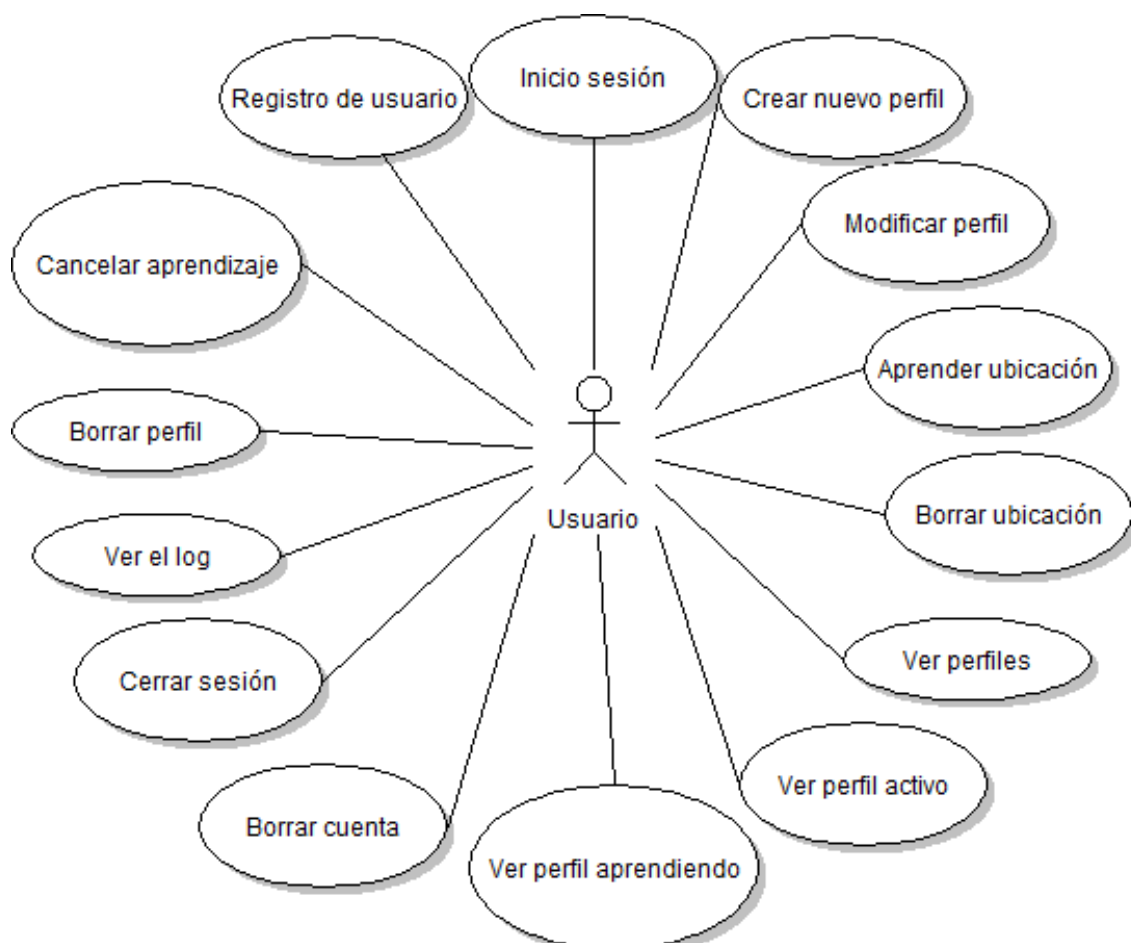


Ilustración 16: Casos de uso

3.4.2 Descripción textual

En este apartado se describen los casos de uso. Para ello se utilizará la siguiente plantilla:

Identificador	
Nombre	
Actores	
Objetivo	
Descripción	
Pre-condición	
Post-condición	

Tabla 24: Plantilla de caso de uso

Dónde cada campo representa lo siguiente:

- **Identificador:** código alfanumérico único que identifica a cada caso de uso. Comienza por CU seguido de un número identificador del caso de uso.
- **Nombre:** identificación descriptiva que recibe el caso de uso. Esta denominación es la que se utiliza en los diagramas UML de casos de uso.
- **Actores:** Se refiere a los roles de las personas o sistemas que podrán desempeñar la función en el sistema utilizando el caso de uso.
- **Objetivo:** establece la finalidad del caso de uso.
- **Descripción:** especificación de los eventos correspondientes a la ejecución estándar del caso de uso.
- **Pre-condición:** especificación detallada de la situación previa a la interacción.
- **Post-condición:** especificación de tallada de la situación posterior a la interacción.

Identificador	CU-01
Nombre	Registro de usuario.
Actores	Usuario.
Objetivo	Registrar un nuevo usuario.
Descripción	<ol style="list-style-type: none">1. El usuario accede a la aplicación.2. El usuario pulsa el botón de registrar usuario.3. El usuario introduce los datos.4. El usuario pulsa el botón registrar.5. La aplicación devuelve que la operación ha sido correcta.
Pre-condición	No existe el usuario que se desea registrar.
Post-condición	Se crea un nuevo usuario.

Tabla 25: CU-01

Identificador	CU-02
Nombre	Inicio de sesión.
Actores	Usuario.
Objetivo	Iniciar sesión dentro de la aplicación.
Descripción	<ol style="list-style-type: none">1. El usuario introduce su nombre de usuario y contraseña.2. El usuario pulsa el botón de Iniciar sesión.3. Si es correcto se inicia sesión en el sistema.
Pre-condición	Se accede a la aplicación desde el dispositivo móvil.
Post-condición	Se accede correctamente a la aplicación.

Tabla 26: CU-02

Identificador	CU-03
Nombre	Nuevo perfil.
Actores	Usuario.
Objetivo	Crear un nuevo perfil.
Descripción	<ol style="list-style-type: none">1. Una vez iniciada sesión, el usuario pulsa el botón de Nuevo perfil.2. El usuario rellena todos los campos.3. El usuario pulsa el botón de Añadir.4. Si los datos son correctos se añade el nuevo perfil.
Pre-condición	El usuario ha iniciado sesión y no existe otro perfil con el mismo nombre.
Post-condición	Se ha creado un nuevo perfil.

Tabla 27: CU-03

Identificador	CU-04
Nombre	Modificar perfil.
Actores	Usuario.
Objetivo	Modificar un perfil existente.
Descripción	<ol style="list-style-type: none">1. Una vez iniciada sesión, el usuario pulsa sobre un perfil existente.2. Selecciona la opción de modificar perfil.3. El usuario rellena los campos del perfil (cambiando solo los que desee).4. El usuario pulsa el botón Modificar.5. Si los datos son correctos el perfil ha modificado sus valores.
Pre-condición	El usuario ha iniciado sesión, el perfil debe existir y no debe existir otro perfil con el nuevo nombre.
Post-condición	Se modifican los datos del perfil.

Tabla 28: CU-04

Identificador	CU-05
Nombre	Aprender ubicación.
Actores	Usuario.
Objetivo	Aprender la ubicación de un perfil.
Descripción	<ol style="list-style-type: none">1. Una vez iniciada sesión, el usuario pulsa sobre un perfil existente.2. Selecciona la opción de Aprender ubicación o Volver a aprender ubicación.3. El usuario selecciona el tiempo por el que quiere que se aprenda.4. Se aprende la ubicación del perfil.
Pre-condición	El usuario ha iniciado sesión y el perfil debe existir (no puede ser el perfil Default).
Post-condición	Se ha aprendido la ubicación del perfil.

Tabla 29: CU-05

Identificador	CU-06
Nombre	Borrar ubicación.
Actores	Usuario.
Objetivo	Borrar la ubicación de un perfil.
Descripción	<ol style="list-style-type: none">1. Una vez iniciada sesión, el usuario pulsa sobre un perfil existente.2. Selecciona la opción de Borrar ubicación.3. Se borra la ubicación del perfil.
Pre-condición	El usuario ha iniciado sesión, el perfil debe existir (no puede ser el perfil Default) y debe tener una ubicación aprendida..
Post-condición	Se ha borrado la ubicación de un perfil.

Tabla 30: CU-06

Identificador	CU-07
Nombre	Borrar perfil.
Actores	Usuario.
Objetivo	Borrar un perfil.
Descripción	<ol style="list-style-type: none">1. Una vez iniciada sesión, el usuario pulsa sobre un perfil existente.2. Selecciona la opción de Borrar perfil.3. Se borra el perfil.
Pre-condición	El usuario ha iniciado sesión y el perfil debe existir.
Post-condición	Se ha borrado un perfil.

Tabla 31: CU-07

Identificador	CU-08
Nombre	Borrar cuenta.
Actores	Usuario.
Objetivo	Borrar una cuenta de usuario.
Descripción	<ol style="list-style-type: none">1. Una vez iniciada sesión, el usuario pulsa el botón Menú.2. Selecciona la opción de Borrar usuario.3. Confirma que desea borrar el usuario.4. Se borra el usuario.
Pre-condición	El usuario ha iniciado sesión.
Post-condición	Se ha borrado una cuenta de usuario.

Tabla 32: CU-08

Identificador	CU-09
Nombre	Cerrar sesión.
Actores	Usuario.
Objetivo	Cerrar la sesión del usuario.
Descripción	<ol style="list-style-type: none">1. Una vez iniciada sesión, el usuario pulsa el botón Menú.2. Selecciona la opción de Cerrar sesión.3. Confirma que desea cerrar la sesión.4. Se cierra la sesión.
Pre-condición	El usuario ha iniciado sesión.
Post-condición	Se cierra la sesión.

Tabla 33: CU-09

Identificador	CU-10
Nombre	Ver el log.
Actores	Usuario.
Objetivo	Ver el log de los perfiles activados.
Descripción	<ol style="list-style-type: none">1. Una vez iniciada sesión, el usuario pulsa el botón Menú.2. Selecciona la opción de Ver log.3. Se carga la lista de los perfiles activados por hora.
Pre-condición	El usuario ha iniciado sesión.
Post-condición	Se muestra la lista de los perfiles activados.

Tabla 34: CU-10

Identificador	CU-11
Nombre	Ver perfiles.
Actores	Usuario.
Objetivo	Ver los perfiles creados.
Descripción	<ol style="list-style-type: none">1. Una vez iniciada sesión, en la pantalla principal se ven los diferentes perfiles.
Pre-condición	El usuario ha iniciado sesión.
Post-condición	Se muestra la lista de los perfiles.

Tabla 35: CU-11

Identificador	CU-12
Nombre	Ver perfil activo.
Actores	Usuario.
Objetivo	Ver el perfil activo
Descripción	<ol style="list-style-type: none">1. Una vez iniciada sesión, en la pantalla principal se ven los diferentes perfiles, y en color verde se muestra el perfil activo.
Pre-condición	El usuario ha iniciado sesión y que exista un perfil activo.
Post-condición	Se muestra la lista de los perfiles, resaltado el perfil activo.

Tabla 36: CU-12

Identificador	CU-13
Nombre	Ver perfil aprendiendo.
Actores	Usuario.
Objetivo	Ver el perfil que está aprendiendo la ubicación.
Descripción	<ol style="list-style-type: none">1. Una vez iniciada sesión, en la pantalla principal se ven los diferentes perfiles, y en color naranja se muestra el perfil que está aprendiendo la ubicación.
Pre-condición	El usuario ha iniciado sesión y que exista un perfil aprendiendo ubicación.
Post-condición	Se muestra la lista de los perfiles, resaltado el perfil que está aprendiendo la ubicación.

Tabla 37: CU-13

Identificador	CU-14
Nombre	Cancelar aprendizaje
Actores	Usuario.
Objetivo	Cancelar el aprendizaje de ubicación de un perfil.
Descripción	<ol style="list-style-type: none">1. Una vez iniciada sesión, en la pantalla principal se ven los diferentes perfiles, y en color naranja se muestra el perfil que está aprendiendo la ubicación.2. Se selecciona el perfil que está aprendiendo.3. Se selecciona la opción Cancelar aprendizaje.4. Se termina el aprendizaje de ese perfil.
Pre-condición	El usuario ha iniciado sesión y que exista un perfil aprendiendo ubicación.
Post-condición	Se cancela el aprendizaje de la ubicación de un perfil.

Tabla

38:

CU-14

3.5 Requisitos Software

En este apartado vamos a detallar los requisitos software, tanto funcionales como no funcionales, obtenido a través del análisis de requisitos de usuario y los casos de uso. Existen dos tipos de requisitos software, los funcionales y los no funcionales. Los requisitos funcionales especifican el “qué” tiene que hacer el software, mientras que los requisitos no funcionales especifican el “cómo” tiene que hacerse y funcionar el software.

El formato seguido para la especificar los requisitos software será el siguiente:

Identificador	
Título	
Prioridad	<input type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Necesidad	<input type="checkbox"/> Esencial <input type="checkbox"/> Conveniente <input type="checkbox"/> Opcional
Estabilidad	<input type="checkbox"/> Sí <input type="checkbox"/> No
Verificable	<input type="checkbox"/> Sí <input type="checkbox"/> No
Fuente	
Descripción	

Tabla 39: Plantilla de requisito de software

Dónde cada campo representa lo siguiente:

- **Identificador:** código alfanumérico único que identifica a cada requisito. Comienza por RS seguido de una F si es funcional o de NF si es no funcional. Para concluir llevan un número identificador del requisito.
- **Título:** es el nombre único del requisito.
- **Prioridad:** hace referencia al orden temporal en que debe realizarse el requisito.
- **Necesidad:** establece la importancia del requisito desde el punto de vista del cliente.
- **Estabilidad:** indica si el requisito se mantendrá igual a lo largo del proyecto o si, por el contrario, podrá ser modificado.
- **Verificable:** indica si se puede comprobar que el requisito está cumplido.
- **Fuente:** indica la procedencia del requisito.
- **Descripción:** detalla la especificación del requisito de manera precisa y simple.

3.5.1 Requisitos de software funcionales

Los requisitos de software funcionales especifican los servicios que la aplicación debe proporcionar.

Identificador	RSF-01
Título	Añadir un nuevo usuario al sistema.
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Conveniente <input type="checkbox"/> Opcional
Estabilidad	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No
Verificable	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No
Fuente	RUC-01
Descripción	Se permitirá a cualquier usuario registrarse en el sistema. Para poder realizar el registro tendrá que rellenar el formulario de registro, que está compuesto por los campos nombre de usuario y contraseña.

Tabla 40: RSF-01

Identificador	RSF-02
Título	Inicio de sesión
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Conveniente <input type="checkbox"/> Opcional
Estabilidad	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No
Verificable	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No
Fuente	RUC-02
Descripción	Se permitirá a cualquier usuario registrado previamente en el sistema iniciar sesión. Para iniciar sesión tendrá que aportar su nombre de usuario y contraseña.

Tabla 41: RSF-02

Identificador	RSF-03
Título	Perfil "Default"
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Conveniente <input type="checkbox"/> Opcional
Estabilidad	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No
Verificable	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No
Fuente	RUC-03
Descripción	Cuando se registra un nuevo usuario el sistema creará un perfil llamado "Default" que se activará cuando no haya ningún otro perfil activo.

Tabla 42: RSF-03

Identificador	RSF-04
Título	Añadir nuevo perfil sin aprendizaje
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Conveniente <input type="checkbox"/> Opcional
Estabilidad	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No
Verificable	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No
Fuente	RUC-04
Descripción	<p>Permitir a cualquier usuario registrado y que ha iniciado sesión crear un nuevo perfil sin que se aprenda la ubicación en el momento. Para crear el nuevo perfil se debe rellenar el formulario correspondiente, con los campos:</p> <ul style="list-style-type: none"> - Nombre. - Wi-Fi. - Bluetooth. - Datos Móviles. - Modo sonido.

Tabla 43: RSF-04

Identificador	RSF-05
Título	Añadir nuevo perfil con aprendizaje
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Conveniente <input type="checkbox"/> Opcional
Estabilidad	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No
Verificable	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No
Fuente	RUC-04
Descripción	<p>Permitir a cualquier usuario registrado y que ha iniciado sesión crear un nuevo perfil y que se aprenda la ubicación en el momento. Para crear el nuevo perfil se debe rellenar el formulario correspondiente, con los campos:</p> <ul style="list-style-type: none"> - Nombre. - Wi-Fi. - Bluetooth. - Datos Móviles. - Modo sonido. - Tiempo de aprendizaje.

Tabla 44: RSF-05

Identificador	RSF-06
Título	Modificar un perfil existente
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Conveniente <input type="checkbox"/> Opcional
Estabilidad	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No
Verificable	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No
Fuente	RUC-05
Descripción	<p>Permitir a cualquier usuario registrado y que ha iniciado sesión modificar un perfil existente. Los datos que se pueden modificar son:</p> <ul style="list-style-type: none"> - Nombre. - Wi-Fi. - Bluetooth. - Datos Móviles. - Modo sonido.

Tabla 45: RSF-06

Identificador	RSF-07
Título	Borrar usuario
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Conveniente <input type="checkbox"/> Opcional
Estabilidad	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No
Verificable	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No
Fuente	RUC-06
Descripción	Permitir a cualquier usuario registrado y que ha iniciado sesión borrar su cuenta de usuario.

Tabla 46: RSF-07

Identificador	RSF-08
Título	Aprender ubicación del perfil
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Conveniente <input type="checkbox"/> Opcional
Estabilidad	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No
Verificable	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No
Fuente	RUC-07
Descripción	Permitir a cualquier usuario registrado y que ha iniciado sesión aprender la ubicación del perfil.

Tabla 47: RSF-08

Identificador	RSF-09
Título	Volver a aprender ubicación del perfil
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Conveniente <input type="checkbox"/> Opcional
Estabilidad	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No
Verificable	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No
Fuente	RUC-07
Descripción	Permitir a cualquier usuario registrado y que ha iniciado sesión volver a aprender la ubicación de un perfil del que ya se ha aprendido previamente la ubicación.

Tabla 48: RSF-09

Identificador	RSF-10
Título	Cancelar aprendizaje ubicación del perfil
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Conveniente <input type="checkbox"/> Opcional
Estabilidad	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No
Verificable	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No
Fuente	RUC-08
Descripción	Permitir a cualquier usuario registrado y que ha iniciado sesión cancelar el aprendizaje de la ubicación de un perfil siempre y cuando se esté realizando en ese momento.

Tabla 49: RSF-10

Identificador	RSF-11
Título	Borrar ubicación perfil
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Conveniente <input type="checkbox"/> Opcional
Estabilidad	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No
Verificable	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No
Fuente	RUC-09
Descripción	Permitir a cualquier usuario registrado y que ha iniciado sesión borrar todas las ubicaciones almacenadas de un perfil en concreto.

Tabla 50: RSF-11

Identificador	RSF-12
Título	Borrar un perfil
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Conveniente <input type="checkbox"/> Opcional
Estabilidad	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No
Verificable	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No
Fuente	RUC-10
Descripción	Permite a cualquier usuario registrado y que ha iniciado sesión borrar un perfil existente.

Tabla 51: RSF-12

Identificador	RSF-13
Título	Visualizar los perfiles
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Conveniente <input type="checkbox"/> Opcional
Estabilidad	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No
Verificable	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No
Fuente	RUC-11
Descripción	Permitir a cualquier usuario registrado y que ha iniciado sesión ver todos los perfiles que previamente creó.

Tabla 52: RSF-13

Identificador	RSF-14
Título	Visualizar perfil activo
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Conveniente <input type="checkbox"/> Opcional
Estabilidad	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No
Verificable	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No
Fuente	RUC-11
Descripción	Permitir a cualquier usuario registrado y que ha iniciado sesión ver el perfil activado.

Tabla 53: RSF-14

Identificador	RSF-15
Título	Visualizar perfil en aprendizaje
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Conveniente <input type="checkbox"/> Opcional
Estabilidad	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No
Verificable	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No
Fuente	RUC-11
Descripción	Permitir a cualquier usuario registrado y que ha iniciado sesión ver si algún perfil está aprendiendo la ubicación.

Tabla 54: RSF-15

Identificador	RSF-16
Título	Visualizar log
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Conveniente <input type="checkbox"/> Opcional
Estabilidad	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No
Verificable	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No
Fuente	RUC-12
Descripción	Permitir a cualquier usuario registrado y que ha iniciado sesión ver el log de los perfiles que se han ido activando, con la fecha y hora en la que lo han hecho.

Tabla 55: RSF-16

Identificador	RSF-17
Título	Cerrar sesión
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Conveniente <input type="checkbox"/> Opcional
Estabilidad	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No
Verificable	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No
Fuente	RUC-13
Descripción	Permitir a cualquier usuario registrado y que ha iniciado sesión cerrar la sesión que ha iniciado.

Tabla 56: RSF-17

Identificador	RSF-18
Título	Activación del perfil
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Conveniente <input type="checkbox"/> Opcional
Estabilidad	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No
Verificable	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No
Fuente	RUC-13
Descripción	Se activará un perfil cuando el sistema detecte que la ubicación en la que se encuentra coincide con una de las guardadas para ese perfil.

Tabla 57: RSF-18

3.5.2 Requisitos de software no funcionales

Los requisitos no funcionales imponen restricciones en el producto desarrollado y en proceso de desarrollo.

Identificador	RSNF-01
Título	Idioma de la aplicación
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Conveniente <input type="checkbox"/> Opcional
Estabilidad	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No
Verificable	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No
Fuente	RUR-01
Descripción	El idioma que tendrá la aplicación por el momento será el castellano, pudiéndose ampliar fácilmente a otros idiomas si en el futuro así se requiriese.

Tabla 58: RSNF-01

Identificador	RSNF-02
Título	Dispositivo necesario
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Conveniente <input type="checkbox"/> Opcional
Estabilidad	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No
Verificable	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No
Fuente	RUR-02
Descripción	La aplicación funcionará únicamente en dispositivos con sistema operativo Android con la versión 2.2 como mínimo.

Tabla 59: RSNF-02

Identificador	RSNF-03
Título	Aprendizaje simultáneos
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Conveniente <input type="checkbox"/> Opcional
Estabilidad	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No
Verificable	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No
Fuente	RUR-04
Descripción	La aplicación no permitirá realizar más de un aprendizaje de ubicación simultáneamente.

Tabla 60: RSNF-03

Identificador	RSNF-04
Título	Sesión persistente
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Conveniente <input type="checkbox"/> Opcional
Estabilidad	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No
Verificable	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No
Fuente	RUR-05
Descripción	Una vez se ha realizado un inicio de sesión, el sistema lo recordará, para evitar al usuario que tenga que volver a introducirlos cada vez que inicie la sesión.

Tabla 61: RSNF-04

Identificador	RSNF-05
Título	Disponibilidad de Wi-Fi
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Conveniente <input type="checkbox"/> Opcional
Estabilidad	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No
Verificable	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No
Fuente	RUR-06
Descripción	Para poder recoger las ubicaciones a través de Wi-Fi es necesario que el dispositivo posea un chip Wi-Fi.

Tabla 62: RSNF-05

Identificador	RSNF-06
Título	Ubicación por Wi-Fi
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Conveniente <input type="checkbox"/> Opcional
Estabilidad	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No
Verificable	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No
Fuente	RUR-07
Descripción	Para poder recoger las ubicaciones a través de Wi-Fi es necesario tener activado la ubicación por Wi-Fi.

Tabla 63: RSNF-06

Identificador	RSNF-07
Título	Determinar localización
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Conveniente <input type="checkbox"/> Opcional
Estabilidad	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No
Verificable	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No
Fuente	RUR-08
Descripción	El sistema utilizará varios métodos para establecer la localización del dispositivo. Estos métodos son: <ul style="list-style-type: none">- Ubicación Wi-Fi: longitud y latitud.- Células GSM.- Router Wi-Fi al que se está conectado, en caso de estarlo.

Tabla 64: RSNF-07

Identificador	RSNF-08
Título	Almacenaje de los usuarios
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Conveniente <input type="checkbox"/> Opcional
Estabilidad	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No
Verificable	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No
Fuente	RUR-03
Descripción	Se almacenará en una base de datos SQLite los datos de los usuarios.

Tabla 65: RSNF-08

Identificador	RSNF-09
Título	Almacenaje de los perfiles
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Conveniente <input type="checkbox"/> Opcional
Estabilidad	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No
Verificable	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No
Fuente	RUR-03
Descripción	Se almacenará en una base de datos SQLite los datos de los perfiles.

Tabla 66: RSNF-09

Identificador	RSNF-10
Título	Almacenaje de las ubicaciones Wi-Fi
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Conveniente <input type="checkbox"/> Opcional
Estabilidad	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No
Verificable	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No
Fuente	RUR-03
Descripción	Se almacenará en una base de datos SQLite los datos de las ubicaciones Wi-Fi del perfil.

Tabla 67: RSNF-10

Identificador	RSNF-11
Título	Almacenaje de las células GSM
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Conveniente <input type="checkbox"/> Opcional
Estabilidad	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No
Verificable	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No
Fuente	RUR-03
Descripción	Se almacenará en una base de datos SQLite los datos de las células GSM del perfil.

Tabla 68: RSNF-11

Identificador	RSNF-12
Título	Almacenaje del punto de acceso Wi-Fi
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Conveniente <input type="checkbox"/> Opcional
Estabilidad	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No
Verificable	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No
Fuente	RUR-03
Descripción	Se almacenará en una base de datos SQLite los datos de los puntos de acceso Wi-Fi del perfil.

Tabla 69: RSNF-12

Identificador	RSNF-13
Título	Almacenaje del log
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Conveniente <input type="checkbox"/> Opcional
Estabilidad	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No
Verificable	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No
Fuente	RUR-03
Descripción	Se almacenará en una base de datos SQLite los datos del log del usuarios.

Tabla 70: RSNF-13

3.6 Matrices de trazabilidad

En este apartado se van a mostrar las matrices de trazabilidad para asegurar que se cumplen los requisitos de software con respecto a los de usuario.

Habr  dos matrices de trazabilidad:

- Matriz de trazabilidad RUC – RSF.
- Matriz de trazabilidad RUR – RSNF.

5.3.1 Matriz de trazabilidad RUC - RSF

RU/RS	RUC-01	RUC-02	RUC-03	RUC-04	RUC-05	RUC-06	RUC-07	RUC-08	RUC-09	RUC-10	RUC-11	RUC-12	RUC-13
RSF-01	X												
RSF-02		X											
RSF-03			X										
RSF-04				X									
RSF-05				X									
RSF-06					X								
RSF-07						X							
RSF-08							X						
RSF-09							X						
RSF-10								X					
RSF-11									X				
RSF-12										X			
RSF-13										X	X		
RSF-14											X		
RSF-15											X		
RSF-16												X	
RSF-17													X
RSF-18													X

Tabla 71: Matriz de trazabilidad RUC - RSF

5.3.2 Matriz de trazabilidad RUR - RSNF

RU/RS	RUC-01	RUC-02	RUC-03	RUC-04	RUC-05	RUC-06	RUC-07	RUC-08
RSNF-01	X							
RSNF-02		X						
RSNF-03				X				
RSNF-04					X			
RSNF-05						X		
RSNF-06							X	
RSNF-07								X
RSNF-08			X					
RSNF-09			X					
RSNF-10			X					
RSNF-11			X					
RSNF-12			X					
RSNF-13			X					

Tabla 72: Matriz de trazabilidad RUC - RSF

4. Diseño del sistema

4.1 Arquitectura del sistema

En este apartado vamos a describir y justificar el diseño arquitectónico que se ha utilizado para desarrollar la aplicación. Se ofrece también una explicación de cada una de las decisiones tomadas a la hora de realizar el diseño.

Como ya se sabe, la aplicación será desarrollada en Android, mientras que la base de datos será una base de datos SQLite incrustada en la aplicación Android, es decir, la base de datos está auto contenida en la aplicación.

Para desarrollar la aplicación se ha utilizado como base el patrón Modelo Vista Controlador.

4.1.1 Modelo Vista Controlador (MVC)

El Modelo Vista Controlador (MVC) es un patrón de desarrollo de software que separa los datos de una aplicación, la interfaz de usuario y la lógica de negocio en tres componentes distintos. El patrón MVC se ve frecuentemente en aplicaciones web, donde la vista es la página HTML y el código que provee de datos dinámicos a la página. El modelo es el sistema de gestión de bases de datos y la lógica de negocio, y el controlador es el responsable de recibir los eventos de entrada desde la vista.

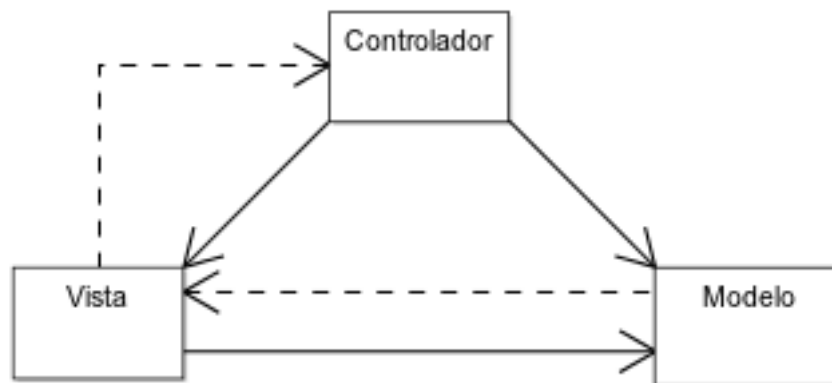


Ilustración 17: Modelo Vista Controlador

El patrón MVC describe una forma de desarrollar aplicaciones software separando los componentes en tres grupos:

- **Modelo:** es la representación específica de la información con la cual el sistema opera. En resumen, el modelo se limita a lo relativo de la vista y su controlador facilitando las presentaciones visuales complejas. El sistema también puede operar con más datos no relativos a la presentación, haciendo uso integrado de

otras lógicas de negocio y de datos afines con el sistema modelado. La parte de la aplicación que forma parte del modelo será la base de datos.

- **Vista:** presenta el modelo en un formato adecuado para interactuar, usualmente la interfaz de usuario. La parte la aplicación que forma parte de la vista son las interfaces de usuario de Android.
- **Controlador:** responde a eventos, usualmente acciones del usuario, e invoca peticiones al modelo y, probablemente, a la vista. La parte de la aplicación que forma parte del controlador son las clases Android que responden a los eventos (por lo general las pulsaciones de los usuarios).

4.2 Diseño detallado

El diseño detallado muestra el diseño de la aplicación. Se mostrará el diseño de las interfaces de usuario Android, el diagrama de clases Android y el diseño de la base de datos. Como se ha visto y explicado anteriormente, todo esto forma parte del Modelo Vista Controlador.

4.2.1 Base de datos

La base de datos de la aplicación contiene todos los datos que guarda y utiliza la aplicación. El diseño de esta base de datos es el siguiente.

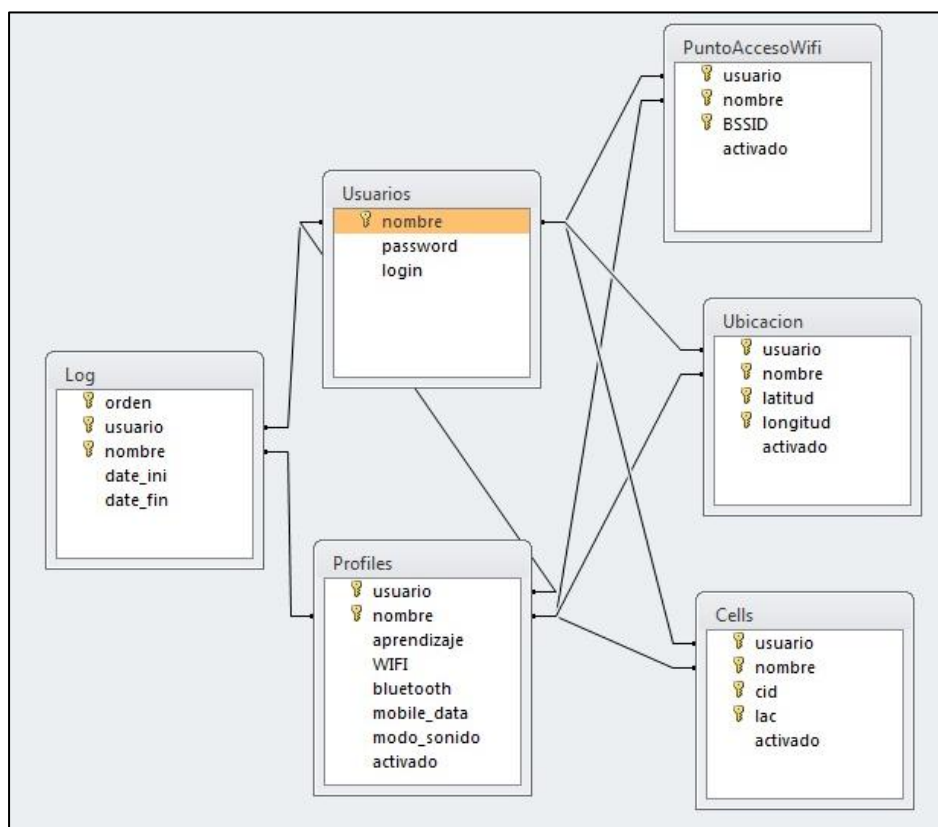


Ilustración 18: Base de datos

A continuación se detallan cada una de estas tablas, utilizando la siguiente plantilla:

Identificador	
Nombre	
Descripción	
Campos	

Tabla 73: Plantilla de tabla de BBDD

Dónde cada campo representa lo siguiente:

- **Identificador:** código alfanumérico que identifica la tabla. Está formado por la letra T y un número identificador del requisito.
- **Nombre:** representa nombre de la tabla.
- **Descripción:** detalla la especificación de la tabla.
- **Campos:** detalla cada uno de los campos de la tabla.

Identificador	T-01
Nombre	Usuarios
Descripción	Representa a los usuarios.
Campos	<ul style="list-style-type: none">- nombre: es el nombre del usuario.- password: es la contraseña del usuario.- login: indica si el usuario ha iniciado sesión.

Tabla 74: T-01

Identificador	T-02
Nombre	Profiles
Descripción	Representa un perfil.
Campos	<ul style="list-style-type: none">- usuario: nombre del usuario al que pertenece el perfil.- nombre: nombre del perfil.- aprendizaje: indica si se está aprendiendo la ubicación del perfil.- WIFI: indica si el Wi-Fi tiene que activarse.- bluetooth: indica si el bluetooth tiene que activarse.- mobile_data: indica si los datos móviles tienen que activarse.- modo_sonido: indica el modo de sonido (normal, vibración o silencio).- activado: indica si el perfil está activo.

Tabla 75: T-02

Identificador	T-03
Nombre	PuntoAccesoWifi
Descripción	Representa un punto de acceso Wi-Fi.
Campos	<ul style="list-style-type: none">- usuario: nombre del usuario al que pertenece el perfil.- nombre: nombre del perfil.- BSSID: BSSID del punto de acceso Wi-Fi.- activado: representa si se está conectado al punto de acceso Wi-Fi que representa.

Tabla 76: T-03

Identificador	T-04
Nombre	Ubicacion
Descripción	Representa una ubicación.
Campos	<ul style="list-style-type: none">- usuario: nombre del usuario al que pertenece el perfil.- nombre: nombre del perfil.- latitud: latitud de la ubicación.- longitud: longitud de la ubicación.- activado: representa si se encuentra en la ubicación que representa.

Tabla 77: T04

Identificador	T-05
Nombre	Cells
Descripción	Representa una célula GSM.
Campos	<ul style="list-style-type: none">- usuario: nombre del usuario al que pertenece el perfil.- nombre: nombre del perfil.- cid: cid de la célula GSM.- lac: lac de la célula GSM.- activado: representa si se encuentra en la célula GSM que representa.

Tabla 78: T-05

Identificador	T-06
Nombre	Log
Descripción	Representa el perfil activado según las horas.
Campos	<ul style="list-style-type: none">- orden: orden de activación del perfil.- usuario: nombre del usuario al que pertenece el perfil.- nombre: nombre del perfil.- date_ini: hora y fecha de activación.- date_fin: hora y fecha de desactivación.

Tabla 79: T-06

4.2.2 Diseño de interfaces de usuario

Al estar desarrollando una aplicación para dispositivos Android, uno de los aspectos más importantes es el diseño de las interfaces de usuario, es decir, de las pantallas que utilizarán los usuarios. Por ello, en este punto se explicarán las decisiones tomadas para el desarrollo de las interfaces.

En una aplicación móvil es muy importante que sea buena gráficamente. El motivo es simple: tiene que tener un diseño lo suficientemente bueno para atraer al público, a los usuarios, para que llegue a usarse y no caiga en el olvido.

Para diseñar las diferentes pantallas de la aplicación hay que tener en cuenta cómo el usuario va a utilizar la aplicación. Para poder comprenderlo se ha realizado en apartados anteriores un estudio de los requisitos, para poder garantizar que las interfaces contengan toda la funcionalidad necesaria.

El objetivo de este apartado es determinar una serie de normas que se han seguido en el desarrollo de las interfaces de usuario, que se mostrarán en el Manual de Usuario, en el anexo B. También se mostrará un diagrama de navegación entre las diferentes pantallas, y se explicará el contenido que las pantallas incorporarán.

Las normas que se han seguido para desarrollar las interfaces son:

- Utilizar interfaces sencillas y cómodas para los usuarios, debido a que si el usuario pierde mucho tiempo en realizar una acción, se pierde eficiencia. Esto ahorrará también mucho tiempo.
- Las diferentes pantallas tendrán un aspecto homogéneo, de forma que se mantenga el mismo color, fuentes y estilos.
- Acceso rápido a las diferentes funcionalidades de la aplicación.
- Los usuarios no tienen por qué ser expertos en informática, por lo que la navegación entre pantallas debe ser sencilla y clara. Por el mismo motivo la aplicación debe ser intuitiva, para ahorrar tiempo de aprendizaje.
- Se debe de seguir los consejos dados por Google en la página web dedicada a desarrolladores.

A partir de estas normas se ha decidido que la aplicación se componga por cinco pantallas. Estas son:

- **Pantalla principal:** esta es la pantalla principal. En esta pantalla se puede realizar el inicio de sesión, rellenando los campos correspondientes. En este caso se accede a la pantalla de vista de perfiles. También se puede navegar a la pantalla de registrar usuario, pulsando el botón correspondiente.
- **Pantalla registrar usuario:** en esta pantalla se puede registrar un nuevo usuario. Para ello se rellenan los campos correspondientes y se registra el usuario. De esta pantalla se navega a la pantalla principal, tanto si se añade el nuevo usuario como si se cancela el registro.
- **Pantalla vista perfiles:** en esta pantalla se mostrará una lista con los diferentes perfiles, mostrando si algún perfil está aprendiendo la ubicación y el perfil que esté activo. Pulsando un botón para añadir perfil se accede a la pantalla añadir perfil. Pulsando cada uno de los perfiles, se pueden modificar, borrar, cancelar aprendizaje y volver a aprender la ubicación. También se puede navegar a la pantalla principal, cerrando sesión, y a la pantalla de log, pulsando la opción correspondiente.

- **Pantalla añadir/modificar perfil:** desde esta pantalla se puede añadir o modificar un perfil, dependiendo de la opción elegida en la pantalla vista de perfiles. Desde esta pantalla se navega siempre a la pantalla vista de perfiles, tanto si se ha añadido o modificado un perfil, como si se cancela la acción.
- **Pantalla de log:** en esta pantalla se muestra una lista con el log de los diferentes perfiles activados según la hora. Desde esta pantalla siempre se volverá a la pantalla vista de perfiles.

A continuación se muestra un pequeño diagrama de navegación entre las diferentes interfaces con lo expuesto anteriormente.

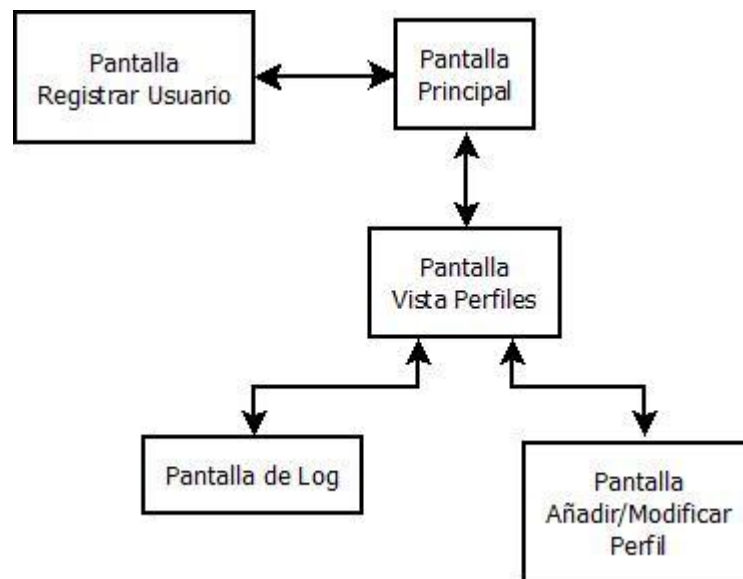


Ilustración 19: Diagrama de navegación

4.2.2 Especificación de clases

En este apartado se centrará en los detalles arquitectónicos de la aplicación. Para ello se explicarán las clases que forman parte de ella, mostrando también un diagrama de clases. Debido a que la aplicación se desarrolla en Android, varias de estas clases se corresponden a las interfaces vistas en el apartado anterior.

Se muestra a continuación el diagrama de clases:

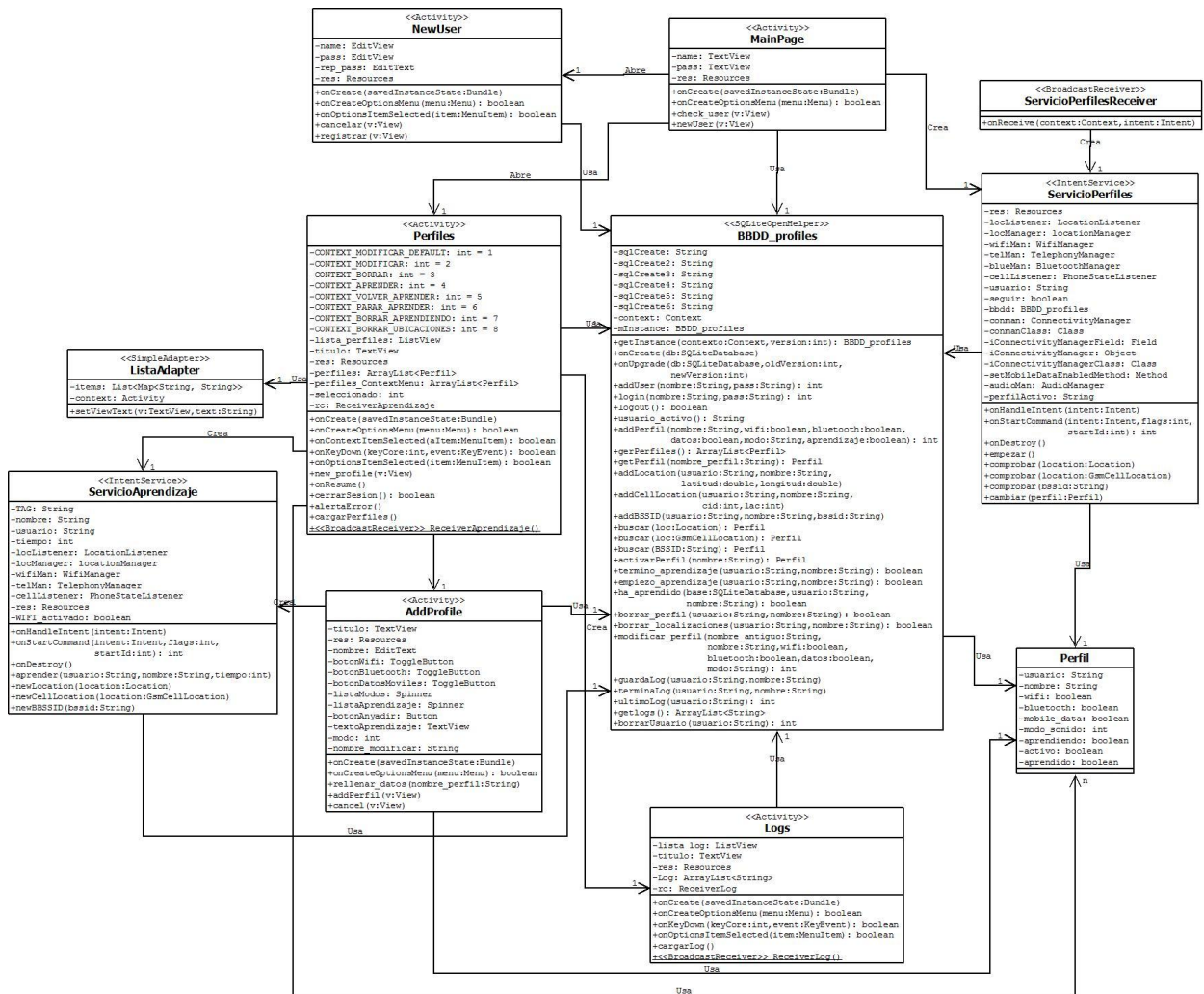


Ilustración 20: Diagrama de clases

De estas clases, varias de ellas se corresponden con las vistas. Estas son:

- Clase MainPage: se corresponde con la vista Pantalla Principal.
- Clase NewUser: se corresponde con la vista Pantalla Registrar Usuario.
- Clase Perfiles: se corresponde con la vista Pantalla Vista Perfiles.
- Clase AddProfile: se corresponde con la vista Pantalla Añadir/Modificar Perfil.
- Clase Logs: se corresponde con la vista Pantalla de Log.

A continuación se explican cada una de las clases que conforman el diagrama. Para explicar las clases se seguirá esta plantilla:

Identificador	
Nombre	
Función	
Dependencias	
Atributos	
Métodos	

Tabla 80: Plantilla de clase

Dónde cada campo representa lo siguiente:

- **Identificador:** código alfanumérico único que identifica a cada clase. Comienza por CL seguido de un número identificador de la clase.
- **Nombre:** representa el nombre de la clase.
- **Función:** indica la función de la clase, lo que hace.
- **Dependencias:** indica las relaciones con otras clases.
- **Atributos:** indica los atributos de la clase.
- **Métodos:** indica los métodos de la clase mediante el formato “Nombre: descripción”.

Identificador	CL-01
Nombre	MainPage
Función	Pantalla de inicio.
Dependencias	NewUser, ServicioPerfiles, Perfiles, BBDD_profiles.
Atributos	name, pass, res.
Métodos	onCreate: método que se ejecuta al crear la pantalla. onCreateOptionsMenu: asigna el menú de opciones. check_user: método que comprueba si el usuario pasado existe y coinciden las contraseñas. Si coincide hace el inicio de sesión y abre la pantalla Perfiles. newUser: método que abre la pantalla NewUser para registrar un usuario.

Tabla 81: CL-01

Identificador	CL-02
Nombre	NewUser
Función	Pantalla para registrar un usuario.
Dependencias	BBDD_profiles.
Atributos	
name, pass, rep_pas, res.	
Métodos	
onCreate: método que se ejecuta al crear la pantalla. onCreateOptionsMenu: asigna el menú de opciones. onOptionsItemSelected: se ejecuta al pulsar una tecla del menú de opciones. cancelar: cancela el registro de usuario, cierra la pantalla. registrar: registra un usuario, se ejecuta al pulsar el botón registrar.	

Tabla 82: CL-02

Identificador	CL-03
Nombre	ServicioPerfilesReceiver
Función	Clase que recibe el evento de fin de encendido del teléfono.
Dependencias	ServicioPerfiles.
Atributos	
Ninguno.	
Métodos	
onReceive: se ejecuta cuando se recibe el evento de fin de encendido del teléfono y ejecuta el servicio ServicioPerfiles.	

Tabla 83: CL-03

Identificador	CL-04
Nombre	ListaAdapter
Función	Clase que define el estilo de la lista de perfiles.
Dependencias	Ninguno.
Atributos	
Ítems, context.	
Métodos	
setViewText: este método define de qué color se ve cada ítem de la lista (verde si está activo, naranja si está aprendiendo o el color por defecto en caso contrario).	

Tabla 84: CL-04

Identificador	CL-05
Nombre	Perfiles
Función	Clase que muestra la lista de perfiles.
Dependencias	ListaAdapter, AddProfile, Perfil, BBDD_profiles, ServicioAprendizaje, Log.
Atributos	
CONTEXT_MODIFICAR_DEFAULT, CONTEXT_MODIFICAR, CONTEXT_BORRAR, CONTEXT_APRENDER, CONTEXT_VOLVER_APRENDER, CONTEXT_BORRAR_APRENDIDO, CONTEXT_BORRAR_UBICACIONES, lista_perfile, titulo, res, perfiles, perfiles_ContextMenu, seleccionado, rc.	
Métodos	
onCreate: método que se ejecuta al crear la pantalla. onCreateOptionsMenu: asigna el menú de opciones. onContextItemSelected: método que se ejecuta al seleccionar un elemento del menú contextual. onKeyDown: método que se ejecuta al pulsar una tecla. onOptionsItemSelected: método que se ejecuta al seleccionar un elemento del menú. New_profile: método que se ejecuta al pulsar el botón Nuevo Perfil, y abre la vista AddProfile. onResume: método que se ejecuta al volver a esta pantalla. cerrarSesion: método que cierra la sesión actual. alertaError: muestra un mensaje de error. cargarPerfiles: carga los diferentes perfiles en la lista. ReceiverAprendizaje: recibe las señales de los servicios para actualizar la lista.	

Tabla 85: CL-05

Identificador	CL-06
Nombre	AddProfile
Función	Clase que sirve para añadir y modificar perfiles
Dependencias	ServicioAprendizaje, BBDD_profiles, Perfil.
Atributos	
Titulo, res, nombre, botonWifi, botonBluetooth, botonDatosMoviles, listaModos, listaAprendizaje, botonAnyadir, textoAprendizaje, modo, nombre_modificar.	
Métodos	
onCreate: método que se ejecuta al crear la pantalla. onCreateOptionsMenu: asigna el menú de opciones. rellenar_datos: rellena la interfaz en el caso de que se esté modificando un perfil. addPerfil: añade o modifica el perfil con los datos de la interfaz. cancel: cierra la interfaz.	

Tabla 86: CL-06

Identificador	CL-07
Nombre	Logs
Función	Clase que muestra el log del usuario.
Dependencias	BBDD_profiles.
Atributos	
Lista_log, titulo, res, Log, rc.	
Métodos	
onCreate: método que se ejecuta al crear la pantalla. onOptionsItemSelected: asigna el menú de opciones. onKeyDown: método que se ejecuta al pulsar una tecla. onOptionsItemSelected: método que se ejecuta al seleccionar un elemento del menú. cargarLog: carga el log y lo introduce en la lista. ReceiverLog: recibe las señales de los servicios para actualizar la lista.	

Tabla 87: CL-07

Identificador	CL-08
Nombre	Perfil
Función	Sirve para definir un perfil.
Dependencias	Ninguna.
Atributos	
Usuario, nombre, wifi, bluetooth, mobile_data, modo_sonido, aprendiendo, activo, aprendido.	
Métodos	
Ninguno.	

Tabla 88: CL-08

Identificador	CL-09
Nombre	ServicioAprendizaje
Función	Clase que aprende la localización de un perfil.
Dependencias	BBDD_profiles.
Atributos	
TAG, nombre, usuario, tiempo, locListener, locManager, wifiMan, telMan, cellListener, res, WIFI_activado.	
Métodos	
onHandleIntent: método que se ejecuta al crearse el servicio. onStartCommand: método que se ejecuta al ejecutar el servicio. onDestroy: método que se ejecuta al terminar el servicio. aprender: método que calcula las nuevas ubicaciones, células GSM y puntos de acceso Wi-Fi del perfil. newLocation: método que guarda en la base de datos una nueva localización del perfil. newCellLocation: método que guarda en la base de datos una nueva célula GSM del perfil. newBSSID: método que guarda en la base de datos un nuevo punto de acceso Wi-Fi del perfil.	

Tabla 89: CL-09

Identificador	CL-10
Nombre	ServicioPerfiles
Función	Clase que aprende la localización de un perfil.
Dependencias	BBDD_profiles, Perfil.
Atributos	
Res, locListener, locManager, wifiMan, telMan, blueMan, cellListener, usuario, seguir, bbdd, conman, conmanClass, iConnectivityManagerField, iConnectivityManager, iConnectivityManagerClass, setMobileDataEnabledMethod, audioMan, perfilActivo.	
Métodos	
onHandleIntent: método que se ejecuta al crearse el servicio. onStartCommand: método que se ejecuta al ejecutar el servicio. onDestroy: método que se ejecuta al terminar el servicio. empezar: método que calcula las ubicaciones, células GSM y puntos de acceso Wi-Fi activas actualmente. comprobar: método que comprueba a qué perfil pertenecen las ubicaciones, células GSM y puntos de acceso Wi-Fi que se encuentran en el método empezar. cambiar: modifica los ajustes del teléfono, poniendo los del perfil activo.	

Tabla 90: CL-10

Identificador	CL-11
Nombre	BBDD_profiles
Función	Clase que representa la base de datos.
Dependencias	Perfil.
Atributos	
sqlCreate, sqlCreate2, sqlCreate3, sqlCreate4, sqlCreate5, sqlCreate6, context, mInstance.	
Métodos	
<p>getInstance: devuelve la instancia de la base de datos (patrón Singleton).</p> <p>onCreate: se ejecuta al crear por primera vez la base de datos.</p> <p>onUpgrade: se ejecuta cuando se cambia la versión de la base de datos.</p> <p>addUser: añade un nuevo usuario.</p> <p>login: modifica la base de datos para indicar que el usuario ha iniciado sesión.</p> <p>logout: modifica la base de datos para indicar que el usuario ya no ha iniciado sesión.</p> <p>usuario_activo: devuelve el usuario activo.</p> <p>addPerfi: añade un nuevo perfil a un usuario.</p> <p>getPerfiles: devuelve la lista de perfiles del usuario activo.</p> <p>getPerfil: devuelve el perfil que coincide con el nombre pasado.</p> <p>addLocation: añade una localización a un perfil.</p> <p>addCellLocation: añade una célula GSM a un perfil.</p> <p>addBSSID: añade un punto de acceso Wi-Fi a un perfil.</p> <p>buscar: busca si la localización, célula GSM o punto de acceso Wi-Fi pertenece a un perfil.</p> <p>activarPerfil: activa un perfil.</p> <p>termino_aprendizaje: modifica la base de datos para indicar que el perfil ya no está aprendiendo.</p> <p>empiezo_aprendizaje: pone un perfil en aprendizaje cambiando el campo correspondiente de la base de datos.</p> <p>ha_aprendido: devuelve si ya se ha aprendido la localización de un perfil.</p> <p>borrar_perfil: borra un perfil.</p> <p>borrar_localizaciones: borra las localizaciones de un perfil.</p> <p>modificar_perfil: modifica un perfil.</p> <p>guardaLog: guarda un nuevo log, con la fecha y hora de inicio.</p> <p>terminaLog: pone la fecha y hora de fin de un nuevo ítem de log.</p> <p>ultimoLog: devuelve la posición del último log.</p> <p>getLogs: devuelve la lista de logs del usuario activo.</p> <p>borrarPerfil: borra el perfil.</p>	

Tabla 91: CL-11

4. Implementación y pruebas

5.1 Implementación

En este apartado se explicará todo el proceso seguido a la hora de realizar la implementación del sistema, así como los diferentes problemas que han surgido y como se han solucionado.

Para poder realizar la aplicación, se han tenido que ir creando los diferentes elementos del sistema. Aquí se muestra el orden que se ha seguido para crear los diferentes elementos:

- Creación de la base de datos: se crea la base de datos que se encuentra dentro de la aplicación Android.
- Creación de las pantallas Android: implementa la vista del usuario, así como la funcionalidad asociada a las pantallas. Estas pantallas se conectan con la base de datos para cumplir su funcionalidad.
- Creación de los servicios de aprendizaje y el servicio principal: se implementan el servicio que aprende la ubicación de un perfil y el servicio que cambia los perfiles dependiendo de la ubicación actual.

5.1.1 Creación de la base de datos

Partiendo del diseño de la base de datos del apartado Diseño del Sistema, se procede a implementar la base de datos como una base de datos SQLite dentro de la aplicación Android. Android nos aporta todas las herramientas necesarias para que una aplicación tenga incrustada una base de datos. La base de datos es la forma más sencilla de tratar con datos en una aplicación Android, por eso se decidió crearla.

Para crear una base de datos en Android, simplemente hay que crear una clase que extiende la clase `SQLiteOpenHelper`. Esta clase proporciona todas las funciones para tratar la base de datos SQLite.

Esta base de datos implementa el patrón Singleton. El patrón de diseño Singleton está diseñado para restringir la creación de objetos pertenecientes a una clase. Su intención consiste en garantizar que una clase sólo tenga una instancia y proporcionar un único punto de acceso global a ella.

Se utiliza el patrón Singleton porque la base de datos se puede llegar a acceder desde varias clases a la vez (además de la interfaz de usuario están los servicios). Utilizando el patrón Singleton evitamos que haya problemas al escribir a la vez en diferentes objetos de la base de datos que en realidad representan a la misma base de datos, es decir, evitamos problemas de concurrencia.

El patrón Singleton se implementa creando en la clase un método que crea una instancia del objeto si no existe una antes. Si existe una antes simplemente se devuelve la

instancia creada anteriormente. Para evitar que el constructor de la clase no se acceda y pueda ser instanciada nuevamente, se regula el alcance del constructor (privado).

En la clase de la base de datos, además de crear las tablas explicadas anteriormente, se crean una serie de métodos para que otras clases puedan operar con la base de datos, limitando así toda interacción con la base de datos a la propia clase. Algunos de estos métodos son: añadir perfil, borrar perfil, modificar perfil, borrar usuario, añadir usuario, etc.

5.1.2 Creación de las pantallas Android

Una vez tenemos creada la base de datos dentro de la aplicación, pasamos a implementar las diferentes pantallas que se explicaron 4.2.2. Estas pantallas son con las que los usuarios de la aplicación interaccionan. A continuación se mostrarán las diferentes pantallas con una breve explicación de las decisiones que se han tomado a la hora de implementarlas.

La primera pantalla de Android implementada es la pantalla principal. Desde esta pantalla se puede iniciar sesión y se puede ir a la pantalla de registro. Se ha decidido incorporar dos campos de texto para el usuario y contraseña, un botón para iniciar sesión y un texto sobre el que se puede pulsar para ir a la pantalla de registro. Se ha decidido que sea un texto en vez de un botón para diferenciarlo del botón de iniciar sesión, y que fuese más pequeño.



Ilustración 21: Pantalla de inicio

Incorpora también controles para que no se dejen los campos vacíos y para que no se inicie sesión si el usuario no existe o la contraseña es errónea.

La siguiente pantalla es la pantalla de registrar. Se ha decidido hacer algo parecido a la pantalla principal, optando por un campo de texto para cancelar el registro. Para introducir el nombre de usuario y contraseña se ha optado por campos de texto.

Al igual que en la pantalla de inicio, existen controles para evitar que se dejen campos vacíos, las contraseñas sean diferentes o se intente registrar un usuario que ya exista en el sistema.

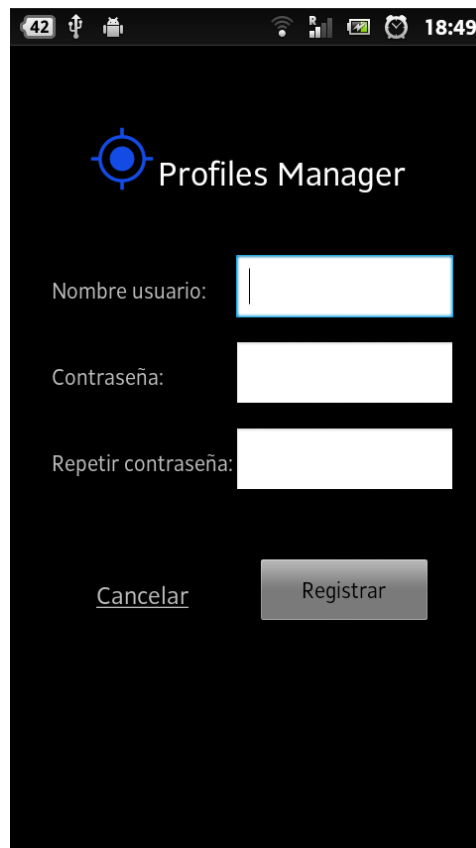


Ilustración 22: Pantalla de registro

En la siguiente pantalla se muestran los diferentes perfiles creados. Se ha optado por una lista para mostrar los perfiles. Se muestra el nombre, y debajo los datos de configuración más importantes: Wi-Fi, datos móviles y el modo de sonido. Para mostrar que un perfil está aprendiendo se ha optado por cambiar el color del texto a naranja, y para mostrar que está activo se muestra el texto en verde.

Para añadir un nuevo perfil se ha optado por un botón. Para modificar perfiles, borrar perfiles o aprender ubicaciones de perfiles se ha optado por utilizar un menú contextual al pulsar un ítem de la lista.

Por último, para abrir la página de log, para borrar el usuario o para cerrar sesión se ha optado por utilizar un menú, desplegado al pulsar la tecla del teléfono correspondiente.

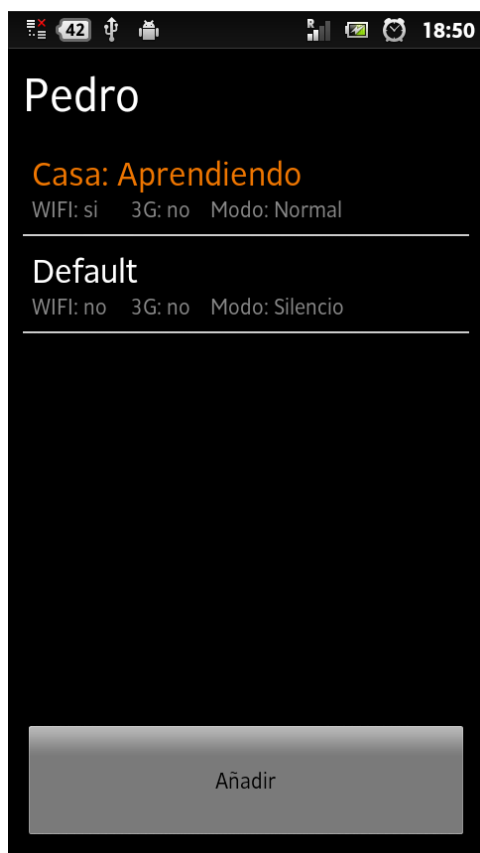


Ilustración 23: Pantalla de perfiles

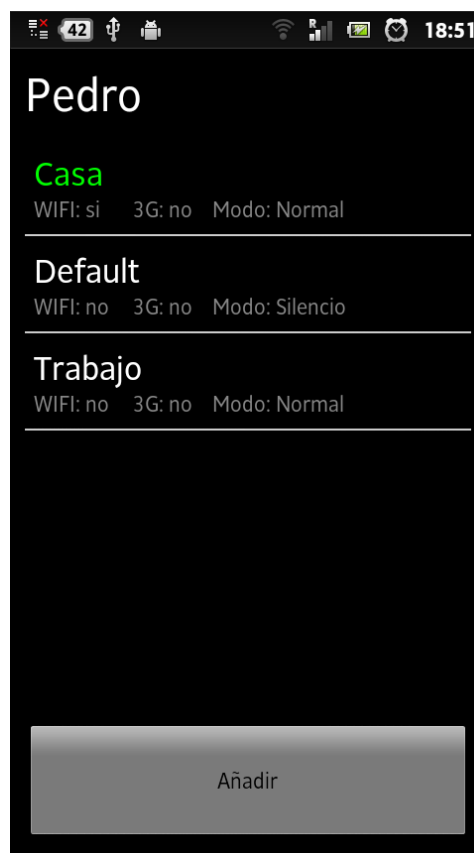


Ilustración 24: Pantalla de perfiles 2

La siguiente pantalla es la que se utiliza para añadir o modificar un perfil. Se ha optado por un campo de texto para el nombre, botones para el Wi-Fi, bluetooth y datos móviles, y una lista desplegable para el modo de sonido y el tiempo de aprendizaje.

Por último, la pantalla para mostrar el log es igual que la pantalla para mostrar los perfiles, mostrando el nombre del perfil activado, con la hora y fecha de inicio y fin. Utilizando el botón atrás se vuelve a la pantalla de los perfiles.

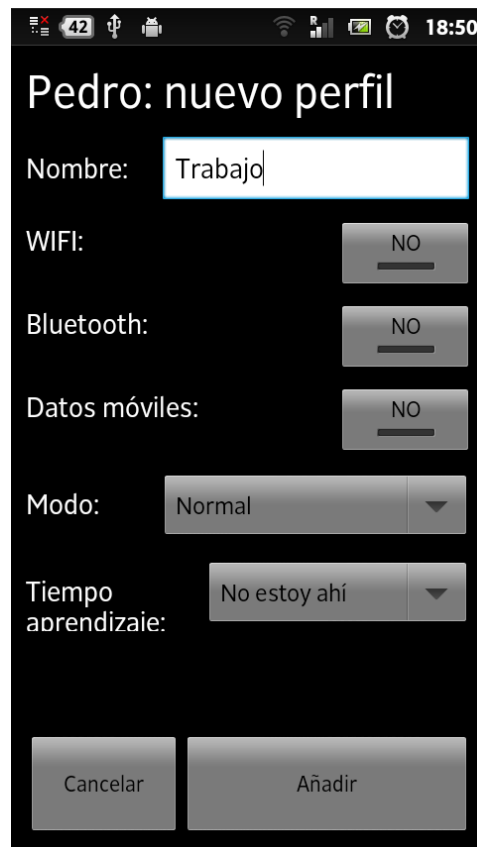


Ilustración 25: Pantalla nuevo/modificar perfil

5.1.3 Creación de los servicios

Dentro de la aplicación existen dos servicios, uno se encargará de aprender la localización para un perfil, y el otro, que se ejecutará en todo momento, se encarga de cambiar la configuración del teléfono dependiendo del perfil activado según la localización.

Ambos eventos extienden a la clase `IntentService`. Esta clase, proporcionada por la API de Android, sirve para manejar peticiones asíncronas. El trabajo que lleva a cabo se ejecuta en segundo plano, a la vez que se pueden estar haciendo otras tareas en la interfaz.

En un principio se pensó en utilizar la clase `Service`, pero esta clase trabaja con peticiones síncronas, lo que hace que se ejecute en el mismo hilo que la interfaz, bloqueándola. Como lo que queremos es que el servicio se ejecute mientras realizamos otras acciones, fue necesario el cambio a `IntentService`.

Ambos servicios funcionan de una forma parecida, pero mientras el servicio de aprendizaje sólo se ejecuta durante un tiempo limitado (decidido por el usuario), el servicio principal se ejecuta en todo momento, siempre y cuando un usuario haya iniciado sesión.

Ambos recogen la información de la ubicación cada cierto tiempo. Recogen tres datos para conocer la ubicación: la localización a través del Wi-Fi (latitud y longitud), las células GSM y el punto de acceso Wi-Fi.

Una vez tienen estos datos, el trabajo que ambos servicios hacen con ellos son diferentes:

- El servicio de aprendizaje los envía a la base de datos para insertarlos como ubicación del perfil en aprendizaje. La base de datos solo los inserta si no existen anteriormente.
- El servicio que se encarga de cambiar la configuración también los envía a la base de datos, pero, en este caso para que compruebe si existe algún perfil al que pertenezca esa localización, célula Wi-Fi o punto de acceso Wi-Fi. La base de datos devuelve un perfil, y el servicio lo activa y cambia la configuración del teléfono.

5.2 Pruebas

En este apartado se muestran una serie de pruebas que se han realizado a la hora de desarrollar el proyecto. Este conjunto de pruebas que se muestran no son las únicas que se han realizado, ya que a lo largo de la implementación se han realizado numerosas pruebas. Aquí sólo mostraremos las pruebas más importantes.

Las pruebas sirven también para demostrar que todos los requisitos software se han cumplido. Por ello se muestra también una matriz de trazabilidad.

5.2.1 Pruebas de aceptación

En este apartado se procede a definir un plan de pruebas que se va a llevar a cabo en el proyecto para asegurar la calidad y el correcto funcionamiento del sistema, así como el cumplimiento de los requisitos.

Para definir las pruebas se utiliza la siguiente plantilla:

Identificador	
Descripción	

Tabla 92: Plantilla de prueba de aceptación

Dónde cada campo representa lo siguiente:

- **Identificador:** código alfanumérico que identifica la prueba. Está formado por las letras PR y un número identificador del requisito.
- **Descripción:** detalla la especificación de la prueba y el resultado esperado.

Identificador	PR-01
Descripción	Se comprueba que un usuario se puede registrar en el sistema.

Tabla 93: PR-01

Identificador	PR-02
Descripción	Se comprueba que surge un error al intentar volver a registrar un usuario que ya existe.

Tabla 94: PR-02

Identificador	PR-03
Descripción	Se comprueba que surge un error al intentar registrar un usuario sin rellenar todos los campos.

Tabla 95: PR-03

Identificador	PR-04
Descripción	Se comprueba que un usuario registrado puede iniciar sesión.

Tabla 96: PR-04

Identificador	PR-05
Descripción	Se comprueba que surge un error al intentar iniciar sesión con un usuario y contraseñas erróneos.

Tabla 97: PR-05

Identificador	PR-06
Descripción	Se comprueba que surge un error al intentar iniciar sesión sin rellenar todos los campos.

Tabla 98: PR-06

Identificador	PR-07
Descripción	Se comprueba al iniciar sesión que existe un perfil llamado Default.

Tabla 99: PR-07

Identificador	PR-08
Descripción	Se comprueba que se puede añadir un nuevo perfil sin aprendizaje rellenando los campos correspondientes.

Tabla 100: PR-08

Identificador	PR-09
Descripción	Se comprueba que se puede añadir un nuevo perfil con aprendizaje rellenando los campos correspondientes.

Tabla 101: PR-09

Identificador	PR-10
Descripción	Se comprueba que surge un error al intentar añadir un nuevo perfil sin rellenar todos sus campos

Tabla 102: PR-10

Identificador	PR-11
Descripción	Se comprueba que se puede modificar un perfil existente, y que éste se modifica correctamente.

Tabla 103: PR-11

Identificador	PR-12
Descripción	Se comprueba que se puede borrar un usuario.

Tabla 104: PR-12

Identificador	PR-13
Descripción	Se comprueba que se puede aprender y volver a aprender la ubicación de un perfil creado.

Tabla 105: PR-13

Identificador	PR-14
Descripción	Se comprueba que se puede cancelar el aprendizaje de un perfil en proceso de aprendizaje.

Tabla 106: PR-14

Identificador	PR-15
Descripción	Se comprueba que se puede borrar las ubicaciones de un perfil.

Tabla 107: PR-15

Identificador	PR-16
Descripción	Se comprueba que se puede borrar un perfil

Tabla 108: PR-16

Identificador	PR-17
Descripción	Se comprueba que se puede visualizar toda la lista de perfiles, y, si existe algún perfil activo o aprendiendo, se muestran en la lista.

Tabla 109: PR-17

Identificador	PR-18
Descripción	Se comprueba que se puede ver el log de los perfiles activados según la fecha.

Tabla 110: PR-18

Identificador	PR-19
Descripción	Se comprueba que se puede cerrar sesión.

Tabla 111: PR-19

Identificador	PR-20
Descripción	Se comprueba que se activa un perfil cuando se está en la localización del perfil.

Tabla 112: PR-20

Identificador	PR-21
Descripción	Se comprueba que el idioma de la aplicación es el castellano

Tabla 113: PR-21

Identificador	PR-22
Descripción	Se comprueba que la aplicación funciona en dispositivos Android con versión a partir de la 2.2.

Tabla 114: PR-22

Identificador	PR-23
Descripción	Se comprueba que la aplicación no permite que se aprenda la ubicación de varios perfiles a la vez.

Tabla 115: PR-23

Identificador	PR-24
Descripción	Se comprueba que la sesión es persistente cerrando y volviendo a abrir la aplicación.

Tabla 116: PR-24

Identificador	PR-25
Descripción	Se comprueba que con el Wi-Fi activado se recogen todo tipo de localizaciones, células GSM y puntos de acceso Wi-Fi.

Tabla 117: PR-25

Identificador	PR-26
Descripción	Se comprueba que se guardan todos los diferentes tipos de datos en la base de datos.

Tabla 118: PR-26

5.2.2 Matriz de trazabilidad

En este apartado se va a mostrar la matriz de trazabilidad entre las pruebas de aceptación y los requisitos de software funcionales, para asegurarse de que las pruebas cubren todos los requisitos.

En la siguiente página se puede ver la matriz de trazabilidad PR-RSF. Como se puede apreciar en la matriz de trazabilidad, con la realización de todas las pruebas especificadas se comprueba que se cumplen todos los requisitos software funcionales.

Las pruebas a partir de la 21 (incluida) están destinadas a comprobar el cumplimiento de los requisitos no funcionales. Por ello, a continuación se muestra la matriz de trazabilidad PR-RSNF para comprobar que se cumplen los requisitos no funcionales.

Matriz de trazabilidad PR-RSF

RSF/P R	RSF -01	RSF -02	RSF -03	RSF -04	RSF -05	RSF -06	RSF -07	RSF -08	RSF -09	RSF -10	RSF -11	RSF -12	RSF -13	RSF -14	RSF -15	RSF -16	RSF -17	RSF -18
PR-01	X																	
PR-02	X																	
PR-03	X																	
PR-04		X																
PR-05		X																
PR-06		X																
PR-07			X															
PR-08				X														
PR-09					X													
PR-10				X	X													
PR-11						X												
PR-12							X											
PR-13								X	X									
PR-14										X								
PR-15											X							
PR-16												X						
PR-17													X	X	X			
PR-18																X		
PR-19																	X	
PR-20																		X

Tabla 119: Matriz de trazabilidad PR-RSF

Matriz de trazabilidad RSNF-PR

RSNF/PR	RSNF-01	RSNF-02	RSNF-03	RSNF-04	RSNF-05	RSNF-06	RSNF-07	RSNF-08	RSNF-09	RSNF-10	RSNF-11	RSNF-12	RSNF-13
PR-21	X												
PR-22		X											
PR-23			X										
PR-24				X									
PR-25					X	X	X						
PR-26								X	X	X	X	X	X

Tabla 120: Matriz de trazabilidad RSNF-PR

6. Conclusiones

5.1 Conclusiones

En este apartado se muestran las conclusiones alcanzadas al desarrollar el proyecto.

Debido al rápido crecimiento del número de Smartphones, ha sido muy interesante el aprendizaje realizado sobre el desarrollo de aplicaciones para dispositivos móviles. El desarrollo de aplicaciones móviles está en auge, y es muy interesante tener los conocimientos necesarios para poder aprovecharlo.

A la hora de desarrollar la aplicación, se han encontrado varios problemas. Estos se describen a continuación:

- Un pequeño problema surgió a la hora de diseñar las interfaces de usuario. La herramienta Eclipse no facilita demasiado el desarrollo de las interfaces. Además, era necesario crear interfaces atractivas y fáciles de utilizar, cosa que no siempre ha sido fácil.
- Otro problema surgió a la hora de crear los servicios para aprender la ubicación y el servicio para cambiar la configuración basándose en la ubicación. Dio muchos problemas a la hora de recoger ubicaciones, aunque finalmente se pudo solucionar.
- Otro problema era el de crear el servicio pero que no gastase mucha batería. Se ha conseguido al final crear un servicio que no afectase considerablemente el gasto de batería.
- Por otro lado estaba el hecho de no contar con ninguna experiencia en el desarrollo de aplicaciones Android, lo que ha requerido un periodo de aprendizaje.

Se han puesto en práctica conocimientos vistos a lo largo de la carrera, como patrones de diseño, bases de datos, diseño de interfaces de usuario... Se ha utilizado también el patrón Modelo Vista Controlador, haciendo la aplicación más efectiva.

Se han aprendido nuevas tecnologías, como el desarrollo de aplicaciones móviles para dispositivos Android. Nunca había desarrollado aplicaciones Android, y ha resultado muy interesante. Las posibilidades son infinitas, y es un mercado muy amplio. Además, con este proyecto se ha conseguido encontrar una solución a un problema que afecta a gran cantidad de usuarios de dispositivos móviles.

5.2 Líneas futuras

En este apartado se explican las tendencias del proyecto fin de carrera realizado con el objetivo de que se pueda continuar en un futuro, mejorando la funcionalidad, pero manteniendo sus ventajas.

La principal mejora que se le podría aplicación sería ampliar las opciones de configuración de un perfil. Se le podrían añadir configuraciones de sonido (mensajes, notificaciones, llamadas, etc), brillo...

Por otro lado se le podría añadir la posibilidad de configurar los perfiles por horas, ya no solo por ubicación. De esta forma se podrían tener, por ejemplo, dos perfiles para cuando se está en casa, uno durante el día, y otro por la noche, sin sonido para que no moleste.

También se podría mejorar el sistema de localización. Actualmente el servicio de localización obtiene las localizaciones cada un periodo de tiempo. Se podría mejorar y cambiar ese periodo de tiempo dependiendo de dónde estemos, si nos estamos moviendo o no, etc. Por ejemplo, la aplicación podría detectar que nos estamos moviendo en coche, y recoger localizaciones cada menos tiempo porque vamos a una alta velocidad.

Otra forma de mejorar el sistema de localización sería introducirle aprendizaje. La aplicación aprende que por las noches, cuando llegamos a casa estamos una cantidad de horas grandes sin salir, por lo que varía el tiempo que tarda en volver a mirar la aplicación. Estos cambios mejorarían el consumo de batería. Además se podría añadir localización GPS para cuando se está en exteriores.

Aunque sería siempre importante mantener la esencia de la aplicación, la aplicación es sencilla e intuitiva, y no se puede arriesgar estos atractivos por intentar meter más funcionalidad, qué, en muchos casos no se aprovecha.

7. Acrónimos y abreviaturas

En este apartado se definen varios acrónimos y abreviaturas que se encuentran a lo largo del documento:

- **RIM:** Research in Motion.
- **GPS:** sistema de posicionamiento global, del inglés Global Position System.
- **GSM:** sistema global para las comunicaciones móviles, del inglés Global System for Mobile .
- **EDGE:** tasas de datos mejoradas para la evolución de GSM, del inglés Enhanced Data Rates for GSM Evolution.
- **HSDPA:** High Speed Downlink Packet Access.
- **LTE:** Long Term Evolution.
- **SMS:** servicio de mensajes cortos, del inglés Short Message Service.
- **MMS:** servicio de mensajes multimedia, del inglés Multimedia Messaging System.
- **POP:** protocolo de oficina de correo, del inglés Post Office Protocol.
- **IMAP:** Internet Message Access Protocol.
- **SMTP:** protocolo para la transferencia simple de correo electrónico, del inglés Simple Mail Transfer Protocol.
- **CDMA:** acceso múltiple por división de código, del inglés Code Division Multiple Access.
- **EVDO:** Evolution-Data Optimized.
- **VPN:** red privada virtual, del inglés Virtual Private Network.
- **WVGA:** Wide Video Graphics Array.
- **USB:** bus universal en serie, del inglés Universal Serial Bus.
- **NFC:** Near Field Communication.
- **MKV:** Matroska Video.
- **AAC:** Advanced Audio Coding.
- **YAFFS:** solo otro sistema de ficheros flash, del inglés Yet Another Flash File System.
- **SQL:** Structured Query Language.

- **ESA:** agencia espacial europea, del inglés European Space Agency.
- **MVC:** Model View Controller.
- **HTML:** HyperText Markup Language.
- **API:** interfaz de programación de aplicaciones, del inglés Application Programming Interface.
- **UML:** lenguaje unificado de modelado, del inglés Unified Modeling Language.

8. Bibliografía

- [1] Android Wikipedia. 2012. Disponible [Internet]:
http://es.wikipedia.org/wiki/Interfaz_de_programaci3n_de_aplicaciones
- [2] Android Developers, 2012. Disponible [Internet]:
<http://developer.android.com/>
- [3] Versiones de Android Wikipedia. 2012. Disponible [Internet]:
http://en.wikipedia.org/wiki/Android_version_history
- [4] Arquitectura de Android. 2011. Disponible [Internet]:
<http://androideity.com/2011/07/04/arquitectura-de-android/>
- [5] SQLite Wikipedia. 2012. Disponible [Internet]:
<http://es.wikipedia.org/wiki/SQLite>
- [6] Aplicaci3n Llama Android. 2012. Disponible [Internet]:
<https://play.google.com/store/apps/details?id=com.kebab.Llama&hl=es>
- [7] Aplicaci3n Locale Android. 2012. Disponible [Internet]:
<https://play.google.com/store/apps/details?id=com.twofortyfouram.locale&hl=es>
- [8] MVC Wikipedia. 2012. Disponible [Internet]:
http://es.wikipedia.org/wiki/Modelo_Vista_Controlador
- [9] Patr3n Singleton Wikipedia. 2012. Disponible [Internet]:
<http://es.wikipedia.org/wiki/Singleton>
- [10] Documentaci3n SQLite. 2012. Disponible [Internet]:
<http://www.sqlite.org/docs.html>

Anexo A: Planificación y presupuesto

1. Planificación

En este apartado se muestra la planificación del proyecto, mostrando las tareas que se han llevado a cabo, así como el diagrama de Grantt de las mismas. El proyecto se ha llevado a cabo a lo largo de casi tres meses.

En el desarrollo se han diferenciado diferentes tareas, que se muestran a continuación.

	i	Nombre de tarea ▼	Duración ▼	Comienzo ▼	Fin ▼
1	✓	<input type="checkbox"/> Sistema de Perfiles por Ubicación	72 días	mié 15/08/12	jue 25/10/12
2	✓	Propuesta del proyecto	6 días	mié 15/08/12	lun 20/08/12
3	✓	<input type="checkbox"/> Análisis del Sistema	19 días	mar 21/08/12	sáb 08/09/12
4	✓	Definición del problema	3 días	mar 21/08/12	jue 23/08/12
5	✓	Requisitos de usuario	6 días	vie 24/08/12	mié 29/08/12
6	✓	Casos de uso	4 días	jue 30/08/12	dom 02/09/12
7	✓	Requisitos software	6 días	lun 03/09/12	sáb 08/09/12
8	✓	<input type="checkbox"/> Diseño del sistema	7 días	dom 09/09/12	sáb 15/09/12
9	✓	Arquitectura del sistema	3 días	dom 09/09/12	mar 11/09/12
10	✓	Diseño detallado	4 días	mié 12/09/12	sáb 15/09/12
11	✓	<input type="checkbox"/> Implementación	26 días	dom 16/09/12	jue 11/10/12
12	✓	Creación de la base de datos	3 días	dom 16/09/12	mar 18/09/12
13	✓	Creación de la aplicación Android	23 días	mié 19/09/12	jue 11/10/12
14	✓	Pruebas	6 días	vie 12/10/12	mié 17/10/12
15	✓	Terminar Documentación	8 días	jue 18/10/12	jue 25/10/12

Ilustración 26: Planificación de tareas

A continuación se muestra el diagrama de Grantt de la planificación. En él se puede ver el desarrollo de las tareas a lo largo del tiempo.

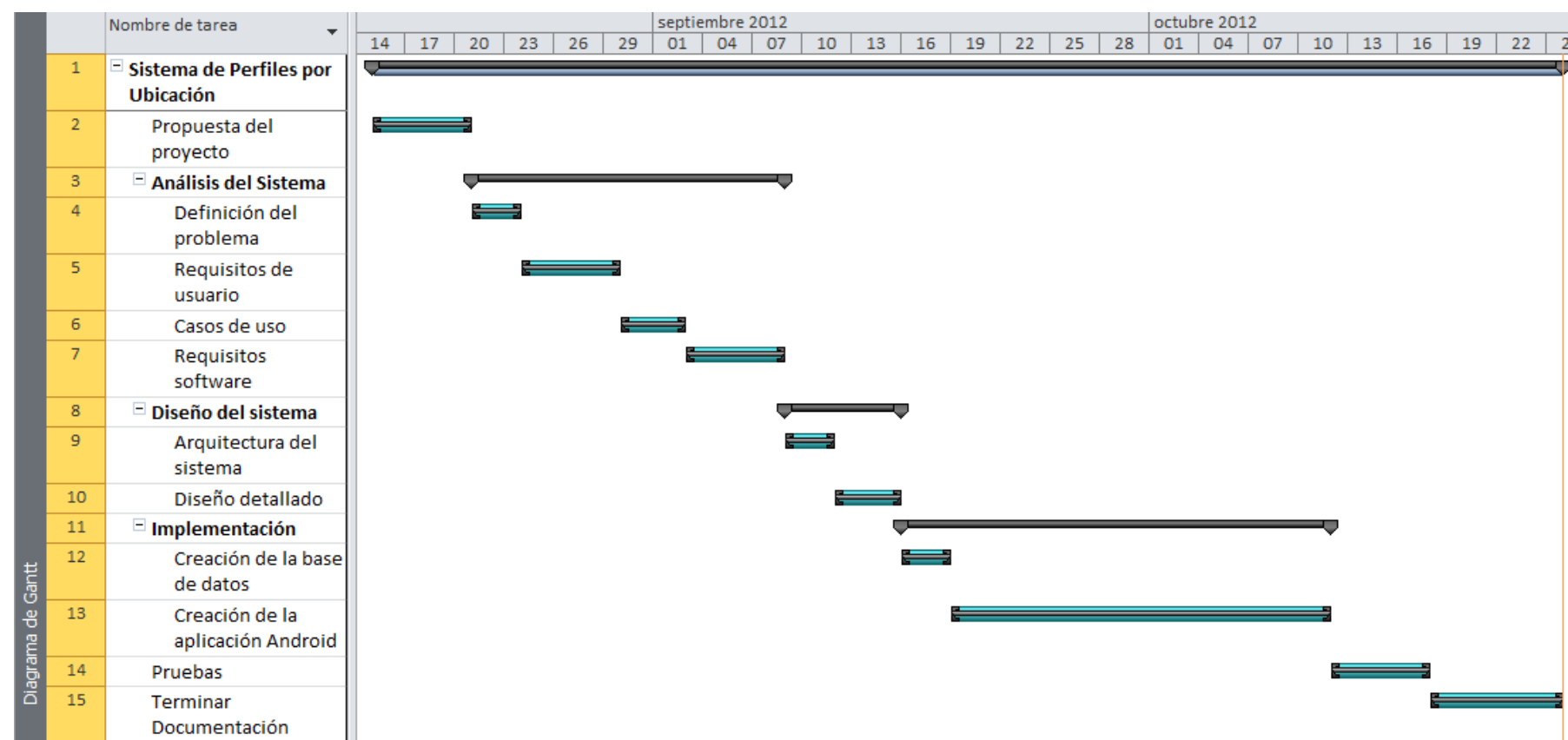


Ilustración 27: Diagrama de Grantt

2. Presupuesto

En este apartado se muestra el presupuesto detallado del proyecto fin de carrera desde su inicio hasta su finalización.

2.1 Coste de personal

Para poder evaluar el coste del personal ha sido necesario evaluar el tiempo dedicado al desarrollo del mismo. Para ello ha sido necesario evaluar el número de días dedicados al proyecto, así como el número de horas al día.

En la siguiente tabla se muestra el personal que ha desarrollado el proyecto. Se ha asumido un coste de 12€/hora, teniendo en cuenta que se ha trabajado una media de 10 horas al día, durante 72 días.

Concepto	Coste por día	Número de días	Coste €
Ingeniero Junior	120	72	8.640
Total			8.640

Tabla 121: Tabla coste de personal

Por lo tanto el coste del personal es de 8.640€.

2.2 Costes de hardware y software

Para poder evaluar los costes de hardware y software ha sido necesario evaluar el cada uno de los elementos y programas utilizados a lo largo del proyecto.

En la siguiente tabla se muestran todos los dispositivos y programas utilizados a lo largo del proyecto.

Concepto	Coste €	Cantidad	Coste total
Sony Xperia U	260	1	260
MacBookAir	1383	1	1.383
Microsoft Office 2011	149	1	149
Microsoft Project 2010	1067	1	1.067
Eclipse	0	1	0
Parallels Desktop 8	79,99	1	79,99
Total			2.938,99

Tabla 122: Tabla de coste de hardware y software

Por lo tanto, el coste total de hardware y software es de 2.938,99€.

2.3 Coste total

En la siguiente tabla se muestra el presupuesto total con todos los costes parciales. Se muestran el precio con y sin IVA, y se aplica además un % de costes indirectos.

Concepto	Coste €	Costes indirectos	Coste total
Coste de personal	8640	10	9.504
Coste de HW/SW	2	10	3.232,89
Total			12.736,89

Tabla 123: Tabla de costes totales

Por lo tanto, el coste total del proyecto sin IVA es de 12.736,89€.

El coste total del proyecto con el IVA del 21% es de **15.411,64€**.

Anexo B: Manual de Usuario

1. Instalación

La aplicación desarrollada tiene una extensión .apk. Es el tipo de archivo de aplicaciones Android. Para poder instalarla en el dispositivo hay que seguir los siguientes pasos:

1. Hay que habilitar el dispositivo para que se puedan instalar aplicaciones de orígenes desconocidos. Para ello nos dirigimos a Ajustes > Aplicaciones y marcamos la opción Orígenes desconocidos.
2. Hay que copiar el fichero .apk en la memoria del teléfono y a través del navegador de archivos hay que ejecutarla e instalarla.

2. Uso de la aplicación

La principal funcionalidad de la aplicación es la de modificar los ajustes del teléfono de acuerdo a perfiles creados y dependiendo de la localización.

La primera pantalla que se va a encontrar, nada más abrir la aplicación es la siguiente:

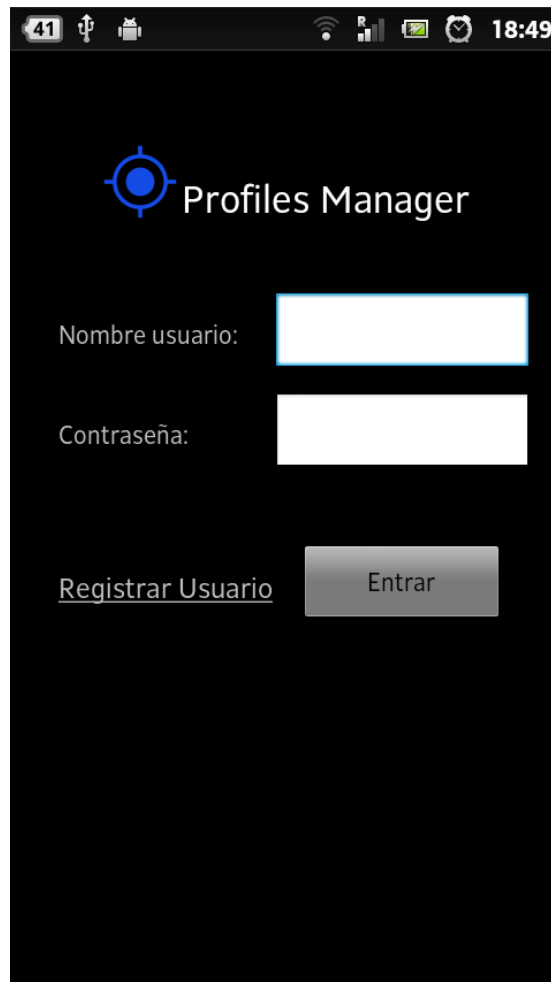


Ilustración 28: Pantalla de inicio de sesión

Desde esta pantalla se pueden realizar dos acciones:

- Iniciar sesión: para ello tendrá que rellenar los campos de Nombre usuario y contraseña y pulsar el botón Entrar.
- Registro de usuario: para ello tendrá que pulsar el texto Registrar Usuario. Le llevará a la pantalla de registrar usuario.

A continuación se muestra la pantalla de registro de usuario.

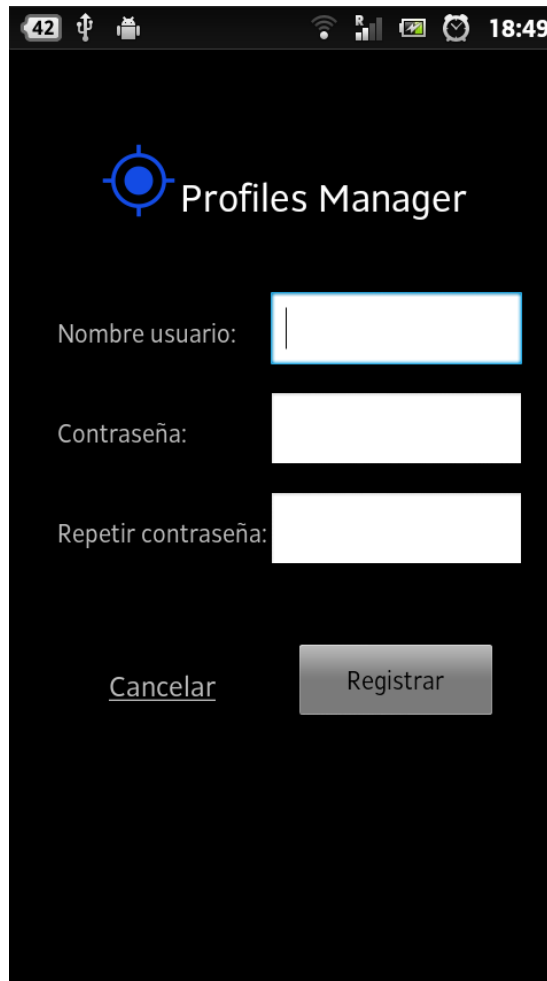


Ilustración 29: Pantalla de registro de usuario

En la pantalla de registrar usuario sólo podrá realizar dos acciones:

- Cancelar y volver a la pantalla de inicio de sesión.
- Registrar un nuevo usuario: para ello deberá rellenar todos los campos. Las dos contraseñas deben de coincidir, y, además el usuario no podrá existir previamente en el sistema.

Una vez hemos iniciado sesión se nos mostrará una pantalla con todos los perfiles creados. Desde esta pantalla se pueden realizar varias acciones:

- Ver los perfiles creados: se muestra una lista de todos los perfiles creados.
- Ver el perfil activo: el perfil activo se muestra en color verde.
- Ver el perfil aprendiendo: si hay algún perfil aprendiendo la ubicación se muestra en color naranja.
- Modificar perfil: para modificar un perfil se presiona prolongadamente hasta que aparece el menú contextual. En el menú contextual se selecciona la opción Modificar, y se modifica el perfil.
- Cancelar aprendizaje: si existe algún perfil aprendiendo la ubicación, se presiona prolongadamente hasta que aparece el menú contextual y se selecciona la opción Cancelar aprendizaje.
- Aprender ubicación: para aprender la ubicación de un perfil se presiona prolongadamente hasta que aparece el menú contextual. En el menú contextual se selecciona la opción Empezar aprendizaje o Volver a aprender. A continuación se selecciona el tiempo de aprendizaje.
- Borrar ubicaciones: si se quieren borrar las ubicaciones aprendidas de un perfil, se pulsa prolongadamente el perfil hasta que aparece el menú contextual. En el menú contextual se selecciona la opción Borrar Ubicaciones.
- Borrar perfil: para borrar un perfil se presiona prolongadamente hasta que aparece el menú contextual. En el menú contextual se selecciona la opción Borrar, y se borra el perfil.
- Crear perfil: para acceder a la pantalla para crear un nuevo perfil hay que pulsar el botón Añadir, que nos llevará a la pantalla de añadir perfiles.
- Cerrar sesión: para cerrar la sesión existen dos opciones: pulsar el botón atrás, o pulsar el botón menú y seleccionar la opción Cerrar sesión.
- Ver log: para acceder a la pantalla de log se tiene que pulsar el botón menú y seleccionar la opción Log.
- Borrar cuenta: para borrar la cuenta se tiene que pulsar el botón menú y seleccionar la opción Borrar cuenta.

A continuación se muestra la pantalla de los perfiles.

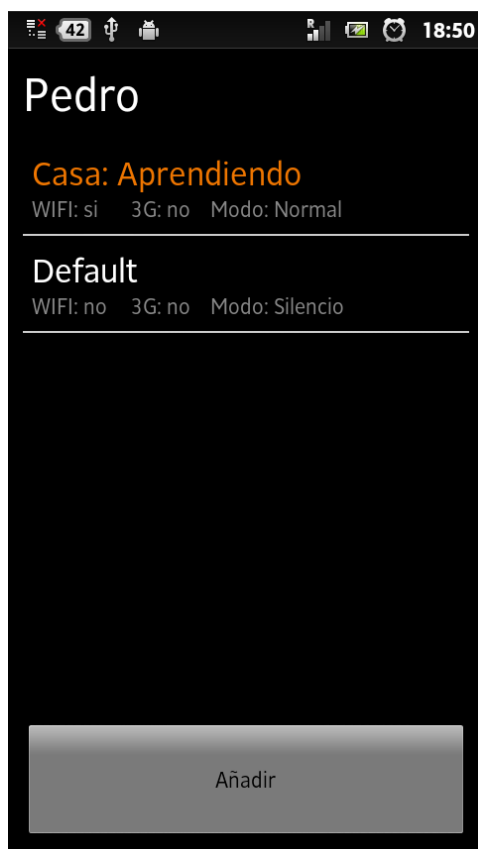


Ilustración 30: Pantalla de perfiles

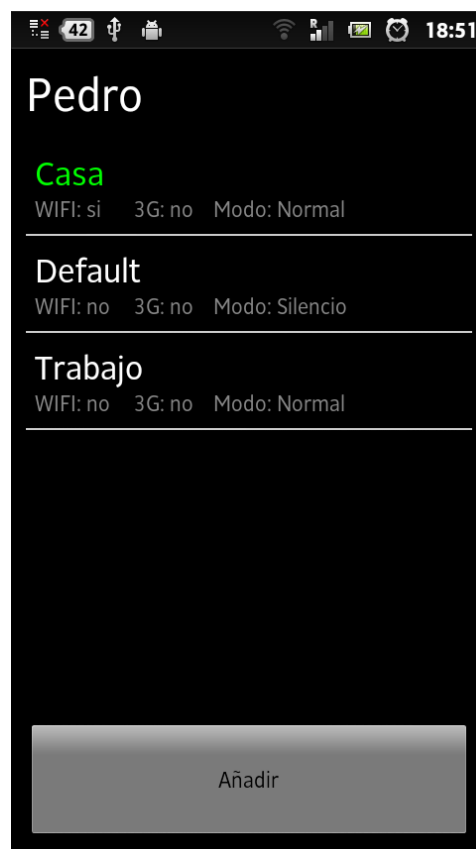


Ilustración 31: Pantalla de perfiles 2

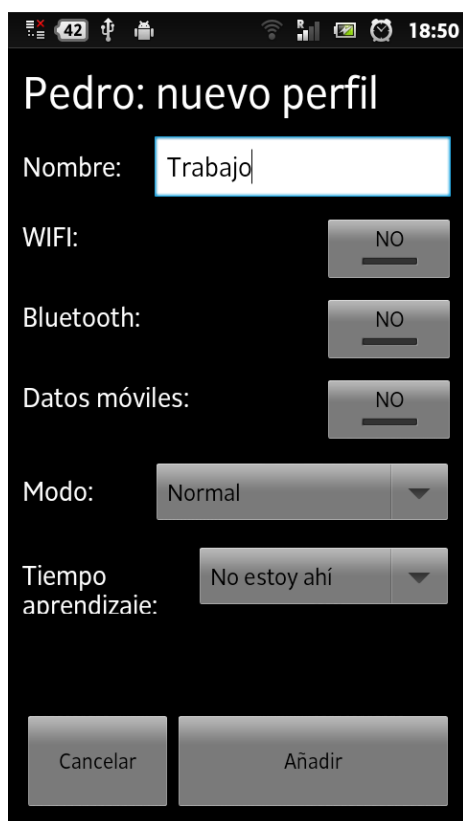


Ilustración 32: Pantalla añadir/modificar perfil

Para añadir o modificar un perfil tendremos que rellenar todos los datos de la pantalla. No podrá existir otro perfil con el mismo nombre.

El tiempo de aprendizaje servirá para que la aplicación aprenda las diferentes ubicaciones del perfil. Para que se aprendan las ubicaciones habrá que seleccionar un tiempo.

Una vez rellenado habrá que pulsar el botón añadir. Si, por lo contrario, se quiere cancelar la inserción o modificación de un perfil se puede pulsar el botón Cancelar.

Los botones No/Si indican si el Wi-Fi, bluetooth o datos móviles estarán activados cuando el perfil se active.

La última pantalla es la pantalla de Log. En esta pantalla se podrán ver las horas en las que se han ido activando los diferentes perfiles. Esta pantalla sirve para conocer las ubicaciones en las que hemos estado.

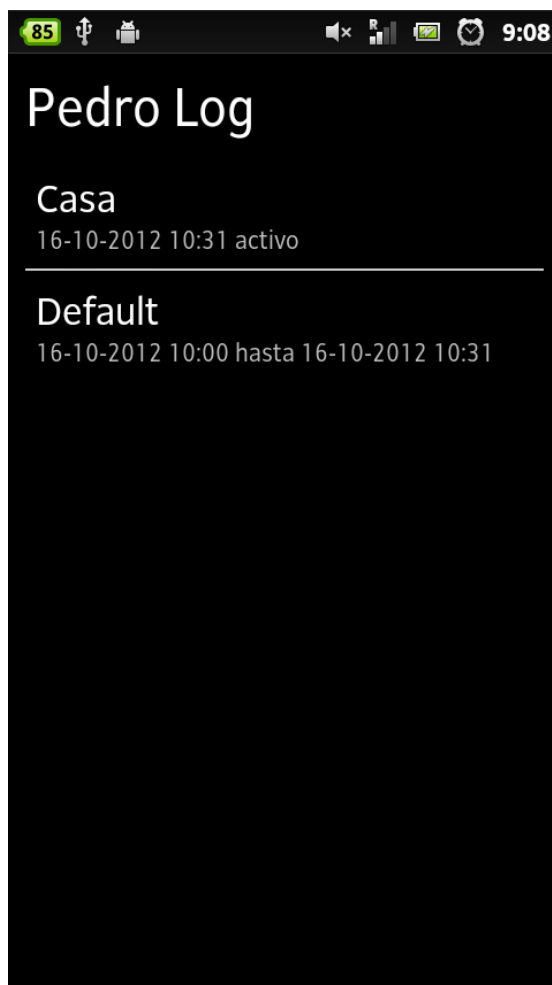


Ilustración 33: Pantalla de Log